Combining Association Measures for Collocation Extraction

Pavel Pecina and **Pavel Schlesinger**
Institute of Formal and Applied Linguistics
Charles University, Prague, Czech Republic
{pecina,schlesinger}@ufal.mff.cuni.cz

Abstract

We introduce the possibility of combining lexical association measures and present empirical results of several methods employed in automatic collocation extraction. First, we present a comprehensive summary overview of association measures and their performance on manually annotated data evaluated by precision--recall graphs and mean average precision. Second, we describe several classification methods for combining association measures, followed by their evaluation and comparison with individual measures. Finally, we propose a feature selection algorithm significantly reducing the number of combined measures with only a small performance degradation.

1 Introduction

Lexical association measures are mathematical formulas determining the strength of association between two or more words based on their occurrences and cooccurrences in a text corpus. They have a wide spectrum of applications in the field of natural language processing and computational linguistics such as automatic collocation extraction (Manning and Schütze, 1999), bilingual word alignment (Mihalcea and Pedersen, 2003) or dependency parsing. A number of various association measures were introduced in the last decades. An overview of the most widely used techniques is given e.g. in Manning and Schütze (1999) or Pearce (2002). Several researchers also attempted to compare existing methods and suggest different evaluation schemes, e.g Kita (1994) and Evert (2001). A comprehensive study of statistical aspects of word cooccurrences can be found in Evert (2004) or Krenn (2000).

In this paper we present a novel approach to automatic collocation extraction based on combining multiple lexical association measures. We also address the issue of the evaluation of association measures by precision-recall graphs and mean average precision scores. Finally, we propose a stepwise feature selection algorithm that reduces the number of combined measures needed with respect to performance on held-out data.

The term *collocation* has both linguistic and lexicographic character. It has various definitions but none of them is widely accepted. We adopt the definition from Choueka (1988) who defines a *collocational expression* as "a syntactic and semantic unit whose exact and unambiguous meaning or connotation cannot be derived directly from the meaning or connotation of its components". This notion of collocation is relatively wide and covers a broad range of lexical phenomena such as idioms, phrasal verbs, light verb compounds, technological expressions, proper names, and stock phrases. Our motivation originates from machine translation: we want to capture all phenomena that may require special treatment in translation.

Experiments presented in this paper were performed on Czech data and our attention was restricted to two-word (*bigram*) collocations – primarily for the limited scalability of some methods to higher-order n-grams and also for the reason that experiments with longer word expressions would require processing of much larger corpus to obtain enough evidence of the observed events.

2 Reference data

The first step in our work was to create a reference data set. Krenn (2000) suggests that collocation extraction methods should be evaluated against a reference set of collocations manually extracted from the full candidate data from a corpus. To avoid the experiments to be biased by underlying data preprocessing (part-of-speech tagging, lemmatization, and parsing), we extracted the reference data from morphologically and syntactically annotated Prague Dependency Treebank 2.0 containing about 1.5 million words annotated on analytical layer (PDT 2.0, 2006). A corpus of this size is certainly not sufficient for real-world applications but we found it adequate for our evaluation purposes – a larger corpus would have made the manual collocation extraction task infeasible.

Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, pages 651–658,
Sydney, July 2006.

Dependency trees from the corpus were broken down into *dependency bigrams* consisting of *lemmas* of the head word and its modifier, their *part-of-speech pattern*, and *dependency type*. From 87 980 sentences containing 1 504 847 words, we obtained a total of 635 952 different dependency bigrams types. Only 26 450 of them occur in the data more than five times. The less frequent bigrams do not meet the requirement of sufficient evidence of observations needed by some methods used in this work (they assume normal distribution of observations and become unreliable when dealing with rare events) and were not included in the evaluation. We, however, must agree with Moore (2004) arguing that these cases comprise majority of all the data (the Zipfian phenomenon) and thus should not be excluded from real-world applications. Finally, we filtered out all bigrams having such part-of-speech patterns that never form a collocation (conjunction–preposition, preposition–pronoun, etc.) and obtained a list consisting of 12 232 dependency bigrams, further called *collocation candidates*.

2.1 Manual annotation

The list of collocation candidates was manually processed by three trained linguists in parallel and independently with the aim of identifying collocations as defined by Choueka. To simplify and clarify the work they were instructed to select those bigrams that can be assigned to these categories:

* idiomatic expressions
  - *studená válka (cold war)*
  - *visí otazník (question mark is hanging ∼ open question)*
* technical terms
  - *předseda vlády (prime minister)*
  - *očitý svědek (eye witness)*
* support verb constructions
  - *mít pravdu (to be right)*
  - *učinit rozhodnutí (make decision)*
* names of persons, locations, and other entities
  - *Pražský hrad (Prague Castle)*
  - *Červený kříž (Red Cross)*
* stock phrases
  - *zásadní problém (major problem)*
  - *konec roku (end of the year)*

The first (expected) observation was that the interannotator agreement among all the categories was rather poor: the Cohen's κ between annotators ranged from 0.29 to 0.49, which demonstrates that the notion of collocation is very subjective, domain-specific, and somewhat vague. The reason that three annotators were used was to get a more precise and objective idea about what can be considered a collocation by combining outcomes from multiple annotators. Only those bigrams that *all* three annotators independently recognized as collocations (of any type) were considered true collocations. The reference data set contains 2 557 such bigrams, which is 20.9% of all. κ between these two categories reanged from 0.52 to 0.58.

The data was split into six stratified samples. Five folds were used for five-fold cross validation and average performance estimation. The remaining one fold was put aside and used as held-out data in experiments described in Section 5.

3 Association measures

In the context of collocation extraction, lexical association measures are formulas determining the degree of association between collocation components. They compute an *association score* for each collocation candidate extracted from a corpus. The scores indicate the potential for a candidate to be a collocation. They can be used for *ranking* (candidates with high scores at the top), or for *classification* (by setting a threshold and discarding all bigrams below this threshold).

If some words occur together more often than by chance, then this may be evidence that they have a special function that is not simply explained as a result of their combination (Manning and Schütze, 1999). This property is known in linguistics as *non-compositionality*. We think of a corpus as a randomly generated sequence of words that is viewed as a sequence of word pairs (dependency bigrams in our case). Occurrence frequencies and marginal frequencies are used in several association measures that reflect how much the word cooccurrence is accidental. Such measures include: estimation of joint and conditional bigram probabilities (Table 1, 1–3), mutual information and derived measures (4–9), statistical tests of independence (10–14), likelihood measures (15–16), and various other heuristic association measures and coefficients (17–55) originating in different research fields.

By determining the entropy of the *immediate context* of a word sequence (words immediately preceding or following the bigram), the association measures (56–60) rank collocations according to the assumption that they occur as (syntactic) units in a (information-theoretically) noisy environment (Shimohata et al., 1997). By comparing *empirical contexts* of a word sequence and of its components (open-class words occurring within

#	Name	Formula
1.	**Joint probability**	$P(xy)$
⋆2.	**Conditional probability**	$P(y\|x)$
3.	**Reverse conditional prob.**	$P(x\|y)$
4.	**Pointwise mutual inform.**	$\log \frac{P(xy)}{P(x*)P(*y)}$
5.	**Mutual dependency (*MD*)**	$\log \frac{P(xy)^2}{P(x*)P(*y)}$
6.	**Log frequency biased *MD***	$\log \frac{P(xy)^2}{P(x*)P(*y)} + \log P(xy)$
7.	**Normalized expectation**	$\frac{2f(xy)}{f(x*)+f(*y)}$
8.	**Mutual expectation**	$\frac{2f(xy)}{f(x*)+f(*y)} \cdot P(xy)$
⋆9.	**Salience**	$\log \frac{P(xy)^2}{P(x*)P(*y)} \cdot \log f(xy)$
10.	**Pearson's χ^2 test**	$\sum_{ij} \frac{(f_{ij}-\hat{f}_{ij})^2}{\hat{f}_{ij}}$
11.	**Fisher's exact test**	$\frac{f(x*)!f(\bar{x}*)!f(*y)!f(*\bar{y})!}{N!f(xy)!f(x\bar{y})!f(\bar{x}y)!f(\bar{x}\bar{y})!}$
12.	**t test**	$\frac{f(xy)-\hat{f}(xy)}{\sqrt{f(xy)(1-(f(xy)/N))}}$
13.	**z score**	$\frac{f(xy)-\hat{f}(xy)}{\sqrt{\hat{f}(xy)(1-(\hat{f}(xy)/N))}}$
14.	**Poison significance measure**	$\frac{f(xy)-f(xy)\log f(xy)+\log f(xy)!}{\log N}$
15.	**Log likelihood ratio**	$-2\sum_{ij} f_{ij} \log \frac{f_{ij}}{\hat{f}_{ij}}$
16.	**Squared log likelihood ratio**	$-2\sum_{ij} \frac{\log f_{ij}^2}{\hat{f}_{ij}}$
	Association coefficients:	
17.	**Russel-Rao**	$\frac{a}{a+b+c+d}$
18.	**Sokal-Michiner**	$\frac{a+d}{a+b+c+d}$
19.	**Rogers-Tanimoto**	$\frac{a+d}{a+2b+2c+d}$
20.	**Hamann**	$\frac{(a+d)-(b+c)}{a+b+c+d}$
21.	**Third Sokal-Sneath**	$\frac{b+c}{a+d}$
22.	**Jaccard**	$\frac{a}{a+b+c}$
⋆23.	**First Kulczynsky**	$\frac{a}{b+c}$
24.	**Second Sokal-Sneath**	$\frac{a}{a+2(b+c)}$
25.	**Second Kulczynski**	$\frac{1}{2}(\frac{a}{a+b}+\frac{a}{a+c})$
⋆26.	**Fourth Sokal-Sneath**	$\frac{1}{4}(\frac{a}{a+b}+\frac{a}{a+c}+\frac{d}{d+b}+\frac{d}{d+c})$
⋆27.	**Odds ratio**	$\frac{ad}{bc}$
28.	**Yulle's ω**	$\frac{\sqrt{ad}-\sqrt{bc}}{\sqrt{ad}+\sqrt{bc}}$
29.	**Yulle's Q**	$\frac{ad-bc}{ad+bc}$
30.	**Driver-Kroeber**	$\frac{a}{\sqrt{(a+b)(a+c)}}$
31.	**Fifth Sokal-Sneath**	$\frac{ad}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$
32.	**Pearson**	$\frac{ad-bc}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$
33.	**Baroni-Urbani**	$\frac{a+\sqrt{ad}}{a+b+c+\sqrt{ad}}$
⋆34.	**Braun-Blanquet**	$\frac{a}{\max(a+b,a+c)}$
⋆35.	**Simpson**	$\frac{a}{\min(a+b,a+c)}$
36.	**Michael**	$\frac{4(ad-bc)}{(a+d)^2+(b+c)^2}$
37.	**Mountford**	$\frac{2a}{2bc+ab+ac}$
38.	**Fager**	$\frac{a}{\sqrt{(a+b)(a+c)}} - \frac{1}{2}\max(b,c)$
39.	**Unigram subtuples**	$\log \frac{ad}{bc} - 3.29\sqrt{\frac{1}{a}+\frac{1}{b}+\frac{1}{c}+\frac{1}{d}}$
40.	***U* cost**	$\log(1+\frac{\min(b,c)+a}{\max(b,c)+a})$
41.	***S* cost**	$\log(1+\frac{\min(b,c)}{a+1})^{-\frac{1}{2}}$
42.	***R* cost**	$\log(1+\frac{a}{a+b})\cdot\log(1+\frac{a}{a+c})$
43.	***T* combined cost**	$\sqrt{U \times S \times R}$
44.	**Phi**	$\frac{P(xy)-P(x*)P(*y)}{\sqrt{P(x*)P(*y)(1-P(x*))(1-P(*y))}}$
45.	**Kappa**	$\frac{P(xy)+P(\bar{x}\bar{y})-P(x*)P(*y)-P(\bar{x}*)P(*\bar{y})}{1-P(x*)P(*y)-P(\bar{x}*)P(*\bar{y})}$
46.	***J* measure**	$\max[P(xy)\log\frac{P(y\|x)}{P(*y)}+P(x\bar{y})\log\frac{P(\bar{y}\|x)}{P(*\bar{y})}, P(xy)\log\frac{P(x\|y)}{P(x*)}+P(\bar{x}y)\log\frac{P(\bar{x}\|y)}{P(\bar{x}*)}]$

#	Name	Formula
47.	**Gini index**	$\max[P(x*)(P(y\|x)^2+P(\bar{y}\|x)^2)-P(*y)^2 - P(\bar{x}*)(P(y\|\bar{x})^2+P(\bar{y}\|\bar{x})^2)-P(*\bar{y})^2, P(*y)(P(x\|y)^2+P(\bar{x}\|y)^2)-P(x*)^2 - P(*\bar{y})(P(x\|\bar{y})^2+P(\bar{x}\|\bar{y})^2)-P(\bar{x}*)^2]$
48.	**Confidence**	$\max[P(y\|x), P(x\|y)]$
49.	**Laplace**	$\max[\frac{NP(xy)+1}{NP(x*)+2}, \frac{NP(xy)+1}{NP(*y)+2}]$
50.	**Conviction**	$\max[\frac{P(x*)P(*y)}{P(x\bar{y})}, \frac{P(\bar{x}*)P(*y)}{P(\bar{x}y)}]$
51.	**Piatersky-Shapiro**	$P(xy)-P(x*)P(*y)$
52.	**Certainity factor**	$\max[\frac{P(y\|x)-P(*y)}{1-P(*y)}, \frac{P(x\|y)-P(x*)}{1-P(x*)}]$
53.	**Added value (*AV*)**	$\max[P(y\|x)-P(*y), P(x\|y)-P(x*)]$
54.	**Collective strength**	$\frac{P(xy)+P(\bar{x}\bar{y})}{P(x*)P(y)+P(\bar{x}*)P(*y)} \cdot \frac{1-P(x*)P(*y)-P(\bar{x}*)P(*y)}{1-P(x*)-P(\bar{x}\bar{y})}$
⋆55.	**Klosgen**	$\sqrt{P(xy)} \cdot AV$
	Context measures:	
⋆56.	**Context entropy**	$-\sum_w P(w\|C_{xy}) \log P(w\|C_{xy})$
⋆57.	**Left context entropy**	$-\sum_w P(w\|C^l_{xy}) \log P(w\|C^l_{xy})$
58.	**Right context entropy**	$-\sum_w P(w\|C^r_{xy}) \log P(w\|C^r_{xy})$
59.	**Left context divergence**	$P(x*)\log P(x*)$ $-\sum_w P(w\|C^l_{xy}) \log P(w\|C^l_{xy})$
60.	**Right context divergence**	$P(*y)\log P(*y)$ $-\sum_w P(w\|C^r_{xy}) \log P(w\|C^r_{xy})$
61.	**Cross entropy**	$-\sum_w P(w\|C_x) \log P(w\|C_y)$
62.	**Reverse cross entropy**	$-\sum_w P(w\|C_y) \log P(w\|C_x)$
63.	**Intersection measure**	$\frac{2\|C_x \cap C_y\|}{\|C_x\|+\|C_y\|}$
⋆64.	**Euclidean norm**	$\sqrt{\sum_w (P(w\|C_x)-P(w\|C_y))^2}$
65.	**Cosine norm**	$\frac{\sum_w P(w\|C_x)P(w\|C_y)}{\sum_w P(w\|C_x)^2 \cdot \sum_w P(w\|C_y)^2}$
⋆66.	***L1* norm**	$\sum_w \|P(w\|C_x)-P(w\|C_y)\|$
67.	**Confusion probability**	$\sum_w \frac{P(x\|C_w)P(y\|C_w)P(w)}{P(x*)}$
⋆68.	**Reverse confusion prob.**	$\sum_w \frac{P(y\|C_w)P(x\|C_w)P(w)}{P(*y)}$
⋆69.	**Jensen-Shannon diverg.**	$\frac{1}{2}[D(p(w\|C_x)\|\|\frac{1}{2}(p(w\|C_x)+p(w\|C_y))) + D(p(w\|C_y)\|\|\frac{1}{2}(p(w\|C_x)+p(w\|C_y)))]$
⋆70.	**Cosine of pointwise *MI***	$\frac{\sum_w MI(w,x)MI(w,y)}{\sqrt{\sum_w MI(w,x)^2}\cdot\sqrt{\sum_w MI(w,y)^2}}$
71.	**KL divergence**	$\sum_w P(w\|C_x) \log \frac{P(w\|C_x)}{P(w\|C_y)}$
72.	**Reverse KL divergence**	$\sum_w P(w\|C_y) \log \frac{P(w\|C_y)}{P(w\|C_x)}$
⋆73.	**Skew divergence**	$D(p(w\|C_x)\|\|\alpha(w\|C_y)+(1-\alpha)p(w\|C_x))$
74.	**Reverse skew divergence**	$D(p(w\|C_y)\|\|\alpha p(w\|C_x)+(1-\alpha)p(w\|C_y))$
75.	**Phrase word coocurrence**	$\frac{1}{2}(\frac{f(x\|C_{xy})}{f(xy)}+\frac{f(y\|C_{xy})}{f(xy)})$
76.	**Word association**	$\frac{1}{2}(\frac{f(x\|C_y)-f(xy)}{f(xy)}+\frac{f(y\|C_x)-f(xy)}{f(xy)})$
	Cosine context similarity:	$\frac{1}{2}(\cos(\mathbf{c}_x,\mathbf{c}_{xy})+\cos(\mathbf{c}_y,\mathbf{c}_{xy}))$ $\mathbf{c}_z=(z_i);\ \cos(\mathbf{c}_x,\mathbf{c}_y)=\frac{\sum x_i y_i}{\sqrt{\sum {x_i}^2}\cdot\sqrt{\sum {y_i}^2}}$
⋆77.	**in boolean vector space**	$z_i = \delta(f(w_i\|C_z))$
78.	**in tf vector space**	$z_i = f(w_i\|C_z)$
79.	**in $tf{\cdot}idf$ vector space**	$z_i = f(w_i\|C_z)\cdot\frac{N}{df(w_i)}$; $df(w_i)=\|\{x : w_i \epsilon C_x\}\|$
	Dice context similarity:	$\frac{1}{2}(\mathrm{dice}(\mathbf{c}_x,\mathbf{c}_{xy})+\mathrm{dice}(\mathbf{c}_y,\mathbf{c}_{xy}))$ $\mathbf{c}_z=(z_i);\ \mathrm{dice}(\mathbf{c}_x,\mathbf{c}_y)=\frac{2\sum x_i y_i}{\sum {x_i}^2+\sum {y_i}^2}$
80.	**in boolean vector space**	$z_i = \delta(f(w_i\|C_z))$
81.	**in tf vector space**	$z_i = f(w_i\|C_z)$
82.	**in $tf{\cdot}idf$ vector space**	$z_i = f(w_i\|C_z)\cdot\frac{N}{df(w_i)}$; $df(w_i)=\|\{x : w_i \epsilon C_x\}\|$

$a=f(xy)$	$b=f(x\bar{y})$	$f(x*)$
$c=f(\bar{x}y)$	$d=f(\bar{x}\bar{y})$	$f(\bar{x}*)$
$f(*y)$	$f(*\bar{y})$	N

A contingency table contains observed frequencies and marginal frequencies for a bigram xy; $\bar{w}$ stands for any word except w; $*$ stands for any word; N is a total number of bigrams. The table cells are sometimes referred to as f_{ij}. Statistical tests of independence work with contingency tables of expected frequencies $\hat{f}(xy) = f(x*)f(*y)/N$.

C_w	empirical context of w
C_{xy}	empirical context of xy
C^l_{xy}	left immediate context of xy
C^r_{xy}	right immediate context of xy

Table 1: Lexical association measures used for bigram collocation extraction.
⋆denotes those selected by the model reduction algorithm discussed in Section 5.

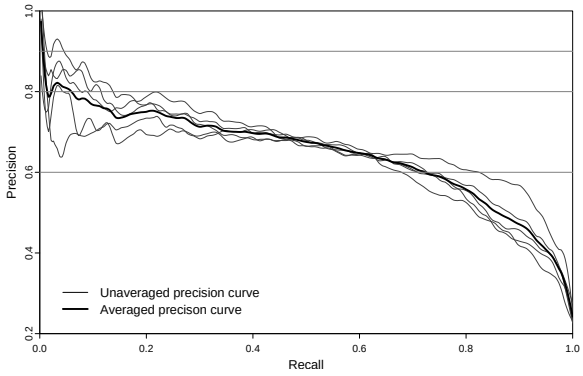

Figure 1: Vertical averaging of precision-recall curves. Thin curves represent individual non-averaged curves obtained by Pointwise mutual information (4) on five data folds.

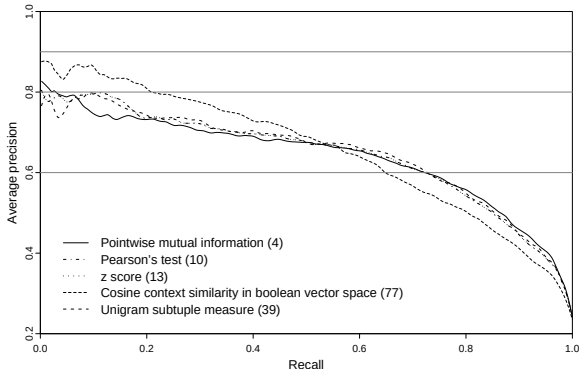

Figure 2: Crossvalidated and averaged precision-recall curves of selected association measures (numbers in brackets refer to Table 1).

a specified context window), the association measures rank collocations according to the assumption that semantically non-compositional expressions typically occur as (semantic) units in different contexts than their components (Zhai, 1997). Measures (61–74) have information theory background and measures (75–82) are adopted from the field of information retrieval.

3.1 Evaluation

Collocation extraction can be viewed as classification into two categories. By setting a threshold, any association measure becomes a binary classifier: bigrams with higher association scores fall into one class (collocations), the rest into the other class (non-collocations). Performance of such classifiers can be measured for example by *accuracy* – fraction of correct predictions. However, the proportion of the two classes in our case is far from equal and we want to distinguish classifier performance between them. In this case, several authors, e.g. Evert (2001), suggest using *precision* – fraction of positive predictions correct and *recall* – fraction of positives correctly predicted. The higher the scores the better the classification is.

3.2 Precision-recall curves

Since choosing a classification threshold depends primarily on the intended application and there is no principled way of finding it (Inkpen and Hirst, 2002), we can measure performance of association measures by precision–recall scores within the entire interval of possible threshold values. In this manner, individual association measures can be thoroughly compared by their two-dimensional *precision-recall curves* visualizing the quality of ranking without committing to a classification threshold. The closer the curve stays to the top and right, the better the ranking procedure is.

Precision-recall curves are very sensitive to data (see Figure 1). In order to obtain a good estimate of their shapes *cross validation* and *averaging* are necessary: all cross-validation folds with scores for each instance are combined and a single curve is drawn. Averaging can be done in three ways: *vertical* – fixing recall, averaging precision, *horizontal* – fixing precision, averaging recall, and *combined* – fixing threshold, averaging both precision and recall (Fawcett, 2003). Vertical averaging, as illustrated in Figure 1, worked reasonably well in our case and was used in all experiments.

3.3 Mean average precision

Visual comparison of precision-recall curves is a powerfull evaluation tool in many research fields (e.g. information retrieval). However, it has a serious weakness. One can easily compare two curves that never cross one another. The curve that predominates another one within the entire interval of recall seems obviously better. When this is not the case, the judgment is not so obvious. Also significance tests on the curves are problematic. Only well-defined one-dimensional quality measures can rank evaluated methods by their performance. We adopt such a measure from information retrieval (Hull, 1993). For each cross-validation data fold we define *average precision* (AP) as the expected value of precision for all possible values of recall (assuming uniform distribution) and *mean average precision* (MAP) as a mean of this measure computed for each data fold. Significance testing in this case can be realized by *paired t-test* or by more appropriate nonparametric *paired Wilcoxon test*.

Due to the unreliable precision scores for low recall and their fast changes for high recall, estimation of AP should be limited only to some narrower recall interval, e.g. $\langle 0.1,0.9 \rangle$

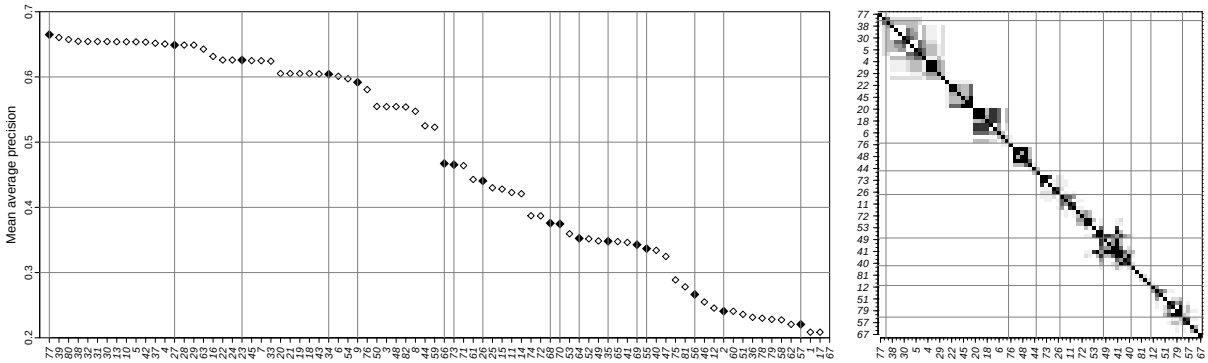

Figure 3: a) Mean average precision of all association measures in descending order. Methods are referred by numbers from Table 1. The solid points correspond to measures selected by the model reduction algorithm from Section 5. b) Visualization of p-values from the significance tests of difference between each method pair (order is the same for both graphs). The darker points correspond to p-values greater than $\alpha = 0.1$ and indicate methods with statistically indistinguishable performance (measured by paired Wilcoxon test on values of average precision obtained from five independent data folds).

3.4 Experiments and results

In the initial experiments, we implemented all 82 association measures from Table 1, processed all morphologically and syntactically annotated sentences from PDT 2.0, and computed scores of all the association measures for each dependency bigram in the reference data. For each association measure and each of the five evaluation data folds, we computed precision-recall scores and drew an averaged precision-recall curve. Curves of some well-performing methods are depicted in Figure 2. Next, for each association measure and each data fold, we estimated scores of average precision on narrower recall interval $\langle 0.1, 0.9 \rangle$, computed mean average precision, ranked the association measures according to MAP in descending order, and result depicted in Figure 3 a). Finally, we applied a paired Wilcoxon test, detected measures with statistically indistinguishable performance, and visualized this information in Figure 3 b).

A baseline system ranking bigrams randomly operates with average precision of 20.9%. The best performing method for collocation extraction measured by mean average precision is *cosine context similarity in boolean vector space* (77) (MAP 66.49%) followed by other 16 association measures with nearly identical performance (Figure 3 a). They include some popular methods well-known to perform reliably in this task, such as *pointwise mutual information* (4), *Pearson's χ^2 test* (10), *z score* (13), *odds ratio* (27), or *squared log likelihood ratio* (16).

The interesting point to note is that, in terms of MAP, context similarity measures, e.g. (77), slightly outperform measures based on simple occurence frequencies, e.g. (39). In a more thorough comparison by percision-recall curves, we observe that the former very significantly predominates the latter in the first half of the recall interval and vice versa in the second half (Figure 2). This is a case where the MAP is not a sufficient metric for comparison of association measure performance. It is also worth pointing out that even if two methods have the same precision-recall curves the actual bigram rank order can be very different. Existence of such *non-correlated* (in terms of ranking) measures will be essential in the following sections.

4 Combining association measures

Each collocation candidate x^i can be described by the *feature vector* $\mathbf{x}^i = (x^i_1, \ldots, x^i_{82})^T$ consisting of 82 association scores from Table 1 and assigned a label $y^i \in \{0, 1\}$ which indicates whether the bigram is considered to be a collocation ($y = 1$) or not ($y = 0$). We look for a *ranker* function $f(\mathbf{x}) \to \mathbb{R}$ that determines the strength of lexical association between components of bigram x and hence has the character of an association measure. This allows us to compare it with other association measures by the same means of precision-recall curves and mean average precision. Further, we present several classification methods and demonstrate how they can be employed for ranking, i.e. what function can be used as a ranker. For references see Venables and Ripley (2002).

4.1 Linear logistic regression

An additive model for binary response is represented by a generalized linear model (GLM) in a form of logistic regression:

$$\mathrm{logit}(\pi) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$$

method	*AP*			*MAP*	
	R=20	*R=50*	*R=80*	*R=⟨0.1,0.9⟩*	*+*
NNet (5 units)	89.56	82.74	70.11	80.81	21.53
NNet (3 units)	89.41	81.99	69.64	79.71	19.88
NNet (2 units)	86.92	81.68	68.33	78.77	18.47
SVM (linear)	85.72	79.49	63.86	75.66	13.79
LDA	84.72	77.18	62.90	75.11	12.96
SVM (quadratic)	84.29	79.54	64.24	74.53	12.09
NNet (1 unit)	77.98	76.83	66.75	73.25	10.17
GLM	82.45	76.26	58.61	71.88	8.11
Cosine similarity (77)	80.94	68.90	50.54	66.49	0.00
Unigram subtuples (39)	74.55	67.49	55.16	65.74	-

Table 2: Performance of methods combining all association measures: average precision (AP) for fixed recall values and mean average precision (MAP) on the narrower recall interval with relative improvement in the last column (values in %).

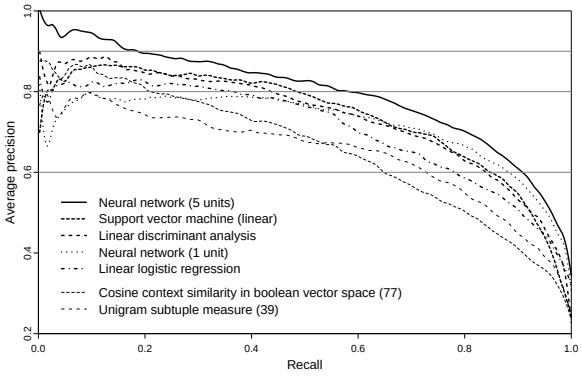

Figure 4: Precision-recall curves of selected methods combining all association measures compared with curves of two best measures employed individually on the same data sets.

where $\text{logit}(\pi) = \log(\pi/(1-\pi))$ is a canonical link function for odds-ratio and $\pi \in (0,1)$ is a conditional probability for positive response given a vector $\mathbf{x}$. The estimation of β_0 and $\boldsymbol{\beta}$ is done by maximum likelihood method which is solved by the *iteratively reweighted least squares algorithm*. The ranker function in this case is defined as the predicted value $\widehat{\pi}$, or equivalently (due to the monotonicity of logit link function) as the linear combination $\widehat{\beta}_0 + \widehat{\boldsymbol{\beta}}^T\mathbf{x}$.

4.2 Linear discriminant analysis

The basic idea of Fisher's linear discriminant analysis (LDA) is to find a one-dimensional projection defined by a vector $\mathbf{c}$ so that for the projected combination $\mathbf{c}^T\mathbf{x}$ the ratio of the *between* variance $\boldsymbol{B}$ to the *within* variance $\boldsymbol{W}$ is maximized:

$$\max_{\mathbf{c}} \frac{\mathbf{c}^T\boldsymbol{B}\mathbf{c}}{\mathbf{c}^T\boldsymbol{W}\mathbf{c}}$$

After projection, $\mathbf{c}^T\mathbf{x}$ can be directly used as ranker.

4.3 Support vector machines

For technical reason, let us now change the labels $y^i \in \{\text{-1,+1}\}$. The goal in support vector machines (SVM) is to estimate a function $f(\mathbf{x}) = \beta_0 + \boldsymbol{\beta}^T\mathbf{x}$ and find a classifier $y(\mathbf{x}) = \text{sign}\big(f(\mathbf{x})\big)$ which can be solved through the following convex optimization:

$$\min_{\beta_0,\boldsymbol{\beta}} \sum_{i=1}^{n} \left[1 - y^i(\beta_0 + \boldsymbol{\beta}^T\,\mathbf{x}^i)\right]^+ + \frac{\lambda}{2}||\boldsymbol{\beta}||^2$$

with λ as a regularization parameter. The *hinge loss function* $L(y, f(\mathbf{x})) = [1 - yf(\mathbf{x})]^+$ is active only for positive values (i.e. bad predictions) and therefore is very suitable for ranking models with $\widehat{\beta}_0 + \widehat{\boldsymbol{\beta}}^T\mathbf{x}$ as a ranker function. Setting the regularization parameter λ is crucial for both the estimators $\widehat{\beta}_0, \widehat{\boldsymbol{\beta}}$ and further classification (or ranking). As an alternative to a often inappropriate grid search, Hastie (2004) proposed an effective algorithm which fits the entire SVM regularization path $[\beta_0(\lambda), \boldsymbol{\beta}(\lambda)]$ and gave us the option to choose the optimal value of λ. As an objective function we used total amount of loss on training data.

4.4 Neural networks

Assuming the most common model of neural networks (NNet) with one hidden layer, the aim is to find inner weights w_{jh} and outer weights w_{hi} for

$$y^i = \phi_0\big(\alpha_0 + \sum w_{hi}\phi_h(\alpha_h + \sum w_{jh}x_j)\big)$$

where h ranges over units in the hidden layer. Activation functions ϕ_h and function ϕ_0 are fixed. Typically, ϕ_h is taken to be the logistic function $\phi_h(z) = \exp(z)/(1+\exp(z))$ and ϕ_0 to be the indicator function $\phi_0(z) = I(z > \Delta)$ with Δ as a classification threshold. For ranking we simply set $\phi_0(z) = z$. Parameters of neural networks are estimated by the *backpropagation algorithm*. The loss function can be based either on *least squares* or *maximum likehood*. To avoid problems with convergence of the algorithm we used the former one. The tuning parameter of a classifier is then the number of units in the hidden layer.

4.5 Experiments and results

To avoid incommensurability of association measures in our experiments, we used a common preprocessing technique for multivariate *standardization*: we centered values of each association measure towards zero and scaled them to unit variance.

Precision-recall curves of all methods were obtained by vertical averaging in five-fold cross validation on the same reference data as in the earlier experiments. Mean average precision was computed from average precision values estimated

on the recall interval $\langle 0.1,0.9 \rangle$. In each cross--validation step, four folds were used for training and one fold for testing.

All methods performed very well in comparison with individual measures. The best result was achieved by a neural network with five units in the hidden layer with 80.81% MAP, which is 21.53% relative improvement compared to the best individual associaton measure. More complex models, such as neural networks with more than five units in the hidden layer and support vector machines with higher order polynomial kernels, were highly overfitted on the training data folds and better results were achieved by simpler models. Detailed results of all experiment are given in Table 2 and precision-recall curves of selected methods depicted in Figure 4.

5 Model reduction

Combining association measures by any of the presented methods is reasonable and helps in the collocation extraction task. However, the combination models are too complex in number of predictors used. Some association measures are very similar (analytically or empirically) and as predictors perhaps even redundant. Such measures have no use in the models, make their training harder, and should be excluded. *Principal component analysis* applied to the evaluation data showed that 95% of its total variance is explained by only 17 principal components and 99.9% is explained by 42 of them. This gives us the idea that we should be able to significantly reduce the number of variables in our models with no (or relativelly small) degradation in their performance.

5.1 The algorithm

A straightforward, but in our case hardly feasible, approach is an exhaustive search through the space of all possible subsets of all association measures. Another option is a heuristic *step-wise* algorithm iteratively removing one variable at a time until some stopping criterion is met. Such algorithms are not very robust, they are sensitive to data and generally not very recommended. However, we tried to avoid these problems by initializing our step-wise algorithm by clustering similar variables and choosing one predictor from each cluster as a representative of variables with the same contribution to the model. Thus we remove the highly corelated predictors and continue with the step--wise procedure.

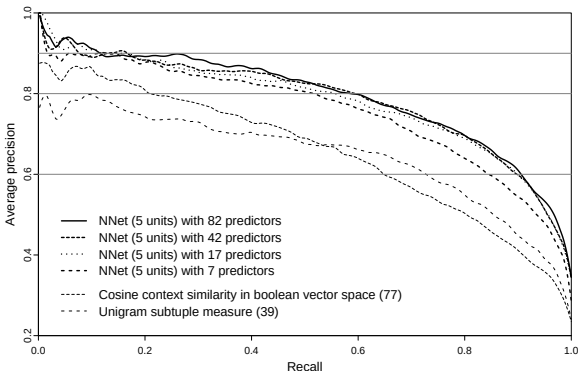

Figure 5: Precision-recall curves of four NNet models from the model reduction process with different number of predictors compared with curves of two best individual methods.

The algorithm starts with the hierarchical clustering of variables in order to group those with a similar contribution to the model, measured by the absolute value of *Pearson's correlation coefficient*. After $82-d$ iterations, variables are grouped into d non-empty clusters and one representative from each cluster is selected as a predictor into the initial model. This selection is based on individual predictor performance on held-out data.

Then, the algorithm continues with d predictors in the initial model and in each iteration removes a predictor causing minimal degradation of performance measured by MAP on held-out data. The algorithm stops when the difference becomes significant – either statistically (by paired Wilcoxon test) or practically (set by a human).

5.2 Experiments and results

We performed the model reduction experiment on the neural network with five units in the hidden layer (the best performing combination method). The similarity matrix for hierarchical clustering was computed on the held-out data and parameter d (number of initial predictors) was experimentally set to 60. In each iteration of the algorithm, we used four data folds (out of the five used in previous experiments) for fitting the models and the held-out fold to measure the performance of these models and to select the variable to be removed. The new model was cross-validated on the same five data-folds as in the previous experiments.

Precision-recall curves for some intermediate models are shown in Figure 5. We can conclude that we were able to reduce the NNet model to about 17 predictors without statistically significant difference in performance. The corresponding association measures are marked in Table 1 and highlighted in Figure 3a). They include measures from the entire range of individual mean average precision values.

6 Conclusions and discussion

We created and manually annotated a reference data set consisting of 12 232 Czech dependency bigrams. 20.9% of them were agreed to be a collocation by three annotators. We implemented 82 association measures, employed them for collocation extraction and evaluated them against the reference data set by averaged precision-recall curves and mean average precision in five-fold cross validation. The best result was achieved by a method measuring *cosine context similarity in boolean vector space* with mean average precision of 66.49%.

We exploit the fact that different subgroups of collocations have different sensitivity to certain association measures and showed that combining these measures aids in collocation extraction. All investigated methods significantly outperformed individual association measures. The best results were achieved by a simple neural network with five units in the hidden layer. Its mean average precision was 80.81% which is 21.53% relative improvement with respect to the best individual measure. Using more complex neural networks or a quadratic separator in support vector machines led to overtraining and did not improve the performace on test data.

We proposed a stepwise feature selection algorithm reducing the number of predictors in combination models and tested it with the neural network. We were able to reduce the number of its variables from 82 to 17 without significant degradation of its performance.

No attempt in our work has been made to select the "best universal method" for combining association measures nor to elicit the "best association measures" for collocation extraction. These tasks depend heavily on data, language, and notion of collocation itself. We demonstrated that combining association measures is meaningful and improves precission and recall of the extraction procedure and full performance improvement can be achieved by a relatively small number of measures combined.

Preliminary results of our research were already published in Pecina (2005). In the current work, we used a new version of the Prague Dependecy Treebank (PDT 2.0, 2006) and the reference data was improved by additional manual anotation by two linguists.

Acknowledgments

This work has been supported by the Ministry of Education of the Czech Republic, projects MSM 0021620838 and LC 536. We would like to thank our advisor Jan Hajič, our colleagues, and anonymous reviewers for their valuable comments.

References

Y. Choueka. 1988. Looking for needles in a haystack or locating interesting collocational expressions in large textual databases. In *Proceedings of the RIAO*.

S. Evert and B. Krenn. 2001. Methods for the qualitative evaluation of lexical association measures. In *Proceedings of the 39th Annual Meeting of the ACL*, Toulouse, France.

S. Evert. 2004. *The Statistics of Word Cooccurrences: Word Pairs and Collocations*. Ph.D. thesis, Univ. of Stuttgart.

T. Fawcett. 2003. ROC graphs: Notes and practical considerations for data mining researchers. Technical report, HPL-2003-4. HP Laboratories, Palo Alto, CA.

T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu. 2004. The entire regularization path for the support vector machine. *Journal of Machine Learning Research*, 5.

D. Hull. 1993. Using statistical testing in the evaluation of retrieval experiments. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, New York, NY.

D. Inkpen and G. Hirst. 2002. Acquiring collocations for lexical choice between near synonyms. In *SIGLEX Workshop on Unsupervised Lexical Acquisition, 40th meeting of the ACL*, Philadelphia.

K. Kita, Y. Kato, T. Omoto, and Y. Yano. 1994. A comparative study of automatic extraction of collocations from corpora: Mutual information vs. cost criteria. *Journal of Natural Language Processing*.

B. Krenn. 2000. *The Usual Suspects: Data-Oriented Models for Identification and Representation of Lexical Collocations*. Ph.D. thesis, Saarland University.

C. D. Manning and H. Schütze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts.

R. Mihalcea and T. Pedersen. 2003. An evaluation exercise for word alignment. In *Proceedings of HLT-NAACL Workshop, Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, Edmonton, Alberta.

R. C. Moore. 2004. On log-likelihood-ratios and the significance of rare events. In *Proceedings of the 2004 Conference on EMNLP*, Barcelona, Spain.

D. Pearce. 2002. A comparative evaluation of collocation extraction techniques. In *Third International Conference on language Resources and Evaluation*, Las Palmas, Spain.

P. Pecina. 2005. An extensive empirical study of collocation extraction methods. In *Proceedings of the ACL 2005 Student Research Workshop*, Ann Arbor, USA.

S. Shimohata, T. Sugio, and J. Nagata. 1997. Retrieving collocations by co-occurrences and word order constraints. In *Proc. of the 35th Meeting of ACL/EACL*, Madrid, Spain.

W. N. Venables and B. D. Ripley. 2002. *Modern Applied Statistics with S. 4th ed.* Springer Verlag, New York.

C. Zhai. 1997. Exploiting context to identify lexical atoms: A statistical view of linguistic context. In *International and Interdisciplinary Conf. on Modeling and Using Context*.

PDT 2.0. 2006. http://ufal.mff.cuni.cz/pdt2.0/.