Lang Resources & Evaluation (2010) 44:137–158
DOI 10.1007/s10579-009-9101-4

Lexical association measures and collocation extraction

Pavel Pecina

Published online: 21 October 2009

Abstract We present an extensive empirical evaluation of collocation extraction methods based on lexical association measures and their combination. The experiments are performed on three sets of collocation candidates extracted from the *Prague Dependency Treebank* with manual morphosyntactic annotation and from the *Czech National Corpus* with automatically assigned lemmas and part-of-speech tags. The collocation candidates were manually labeled as *collocational* or *non-collocational*. The evaluation is based on measuring the quality of ranking the candidates according to their chance to form collocations. Performance of the methods is compared by *precision-recall* curves and *mean average precision* scores. The work is focused on two-word (bigram) collocations only. We experiment with bigrams extracted from sentence dependency structure as well as from surface word order. Further, we study the effect of corpus size on the performance of the individual methods and their combination.

Keywords Lexical association measures · Collocations · Multiword expressions · Evaluation

1 Introduction

The process of combining words into phrases and sentences of natural language is governed by a complex system of rules and constraints. In general, basic rules are given by syntax, however there are also other restrictions (semantic and pragmatic) that must be adhered to in order to produce correct, meaningful, and fluent utterances. These constrains form important linguistic and lexicographic phenomena generally denoted by the term *collocations*. They range from lexically restricted

P. Pecina (✉)
Institute of Formal and Applied Linguistics, Charles University, Prague, Czech Republic
e-mail: pecina@ufal.mff.cuni.cz

expressions (*strong tea, broad daylight*), phrasal verbs (*switch off, look after*), technical terms (*car oil, stock owl*), and proper names (*New York, Old Town*), to idioms (*kick the bucket, hear through the grapevine*), etc. As opposed to free word combinations, collocations are not entirely predictable only on the basis of syntactic rules, they should be listed in a lexicon and learned in the same way as single words are (Palmer 1938).

There is no precise and commonly accepted definition of collocations. Our notion of this phenomenon is based on the definition by Choueka (1988) saying that "[A collocation expression] has a characteristics of a syntactic and semantic unit whose exact and unambiguous meaning or connotation cannot be derived directly from the meaning or connotation of its components." It is relatively wide and covers all unpredictable expressions. This unpredictability is the reason why they should be extensionally specified (listed) in the lexicon. Similar approach is also used by Evert (2004) who defines collocation directly as "a word combination whose semantic and/or syntactic properties cannot be fully predicted from those of its components, and which therefore has to be listed in a lexicon" (Evert 2004, p. 9).

Automatic acquisition of collocations for such a lexicon is one of the traditional tasks of corpus linguistics. The goal is to extract a list of collocations from a text corpus. Generally, it is not required to identify particular occurrences (instances, tokens) of collocations, but rather to produce a list of all collocations (types) appearing anywhere in the corpus. The task is often restricted to a particular subtype or subset of collocations (defined e.g. by grammatical constraints), but we will deal with it in a general sense.

Most of the methods for collocation extraction are based on *lexical association measures* – mathematical formulas determining the strength of association between two or more words based on their occurrences and cooccurrences in a text corpus. The higher the association between words, the better chance they form a collocation. The first research attempts in this area are dated back to the era of *mechanized documentation* (Stevens et al. 1965). The first work focused particularly on collocation extraction was published by Berry-Rogghe (1973), and later followed by studies by Choueka et al. (1983), Church and Hanks (1990), Smadja (1993), Kita et al. (1994), Daille (1996), Shimohata et al. (1997), and many others, especially in the last 10 years (Krenn 2000; Evert 2004; Bartsch 2004).

In the last decades, a number of various association measures have been introduced. An overview of the most widely used techniques is given e.g. in (Manning and Schütze 1999) or (Pearce 2002). Several researchers have also attempted to compare existing methods and suggest different evaluation schemes, e.g. Kita et al. (1994) and Evert and Krenn (2001). A comprehensive study of statistical aspects of word cooccurrences can be found in Evert (2004) or Krenn (2000).

In this work, we study collocation extraction methods based on individual association measures and also on their combination proposed in our previous work (Pecina and Schlesinger 2006). Our evaluation scheme is based on measuring the quality of *ranking* the candidates according to their chance to form collocations. Performance of the methods is compared by *precision-recall* curves and *mean average precision* scores. Our experiments are performed on Czech data and our attention is 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 a much larger corpus to obtain enough evidence of the observed events.

2 Reference data

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 (e.g. all occurring bigrams). However, we limit ourselves only on bigrams occurring in the corpus more than five times (*frequency filter*). 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 are not included in the evaluation, even though we agree with Moore (2004) arguing that these cases comprise majority of all the data (the well-known Zipf phenomenon) and should not be excluded from real-world applications. Further, we filter out all bigrams having such part-of-speech patterns that never form a collocation (*part-of-speech filter*), such as *conjunction–preposition*, *preposition–pronoun*, etc. While designing our experiments and creating the evaluation data sets we proceed with the following three scenarios:

PDT-Dep. To avoid experimental bias from the underlying data preprocessing (part-of-speech tagging, lemmatization, and parsing) necessary for morphologically rich languages such as Czech, we attempt to extract collocations as *dependency bigrams* (not-necessarily contiguous word pairs consisting of a head word and its modifier) from morphologically and syntactically annotated *Prague Dependency Treebank 2.0* (PDT 2006) containing about 1.5 million words annotated on the analytical layer. After applying the frequency and part-of-speech pattern filter, we obtain a set of 12,232 collocation candidates (consisting of lemmas of the head word and its modifier, their part-of-speech pattern, and dependency type) further referred to as *PDT-Dep*.

PDT-Surf. Although collocations form syntactic units by definition, we also attempt to extract collocations from the annotated PDT as *surface bigrams* (pairs of adjacent words) without guarantee that they form such units but with the assumption that majority of bigram collocations can not be modified by insertion of another word and in text they occur as surface bigrams (Manning and Schütze 1999). This approach does not require the source corpus to be parsed, which is usually a time-consuming process, accurate only to a certain extent. After applying the filters, we obtain a set of 10,021 collocation candidates (consisting of component lemmas and their part-of-speech pattern) further referred to as *PDT-Surf*. 974 of these bigrams do not appear in the *PDT-Dep* set (when ignoring syntactic information).

CNC-Surf. A corpus the size of PDT is certainly not sufficient for real-world applications. A larger source corpus would provide not only a greater quantity of collocation candidates (and collocations themselves) but also a better quality of estimates of their frequency characteristics. In order to study the effect of using

Table 1 Summary statistics of the three reference data sets and the source corpora they were extracted from

Reference data set	*PDT-Dep*	*PDT-Surf*	*CNC-Surf*
Sentences	87,980	87,980	15,934,590
Tokens	1,504,847	1,504,847	242,272,798
All bigram types	635,952	638,030	30,608,916
After frequency filtering	26,450	29,035	2,941,414
After part-of-speech filtering	12,232	10,021	1,503,072
Collocation candidates	12,232	10,021	9,868
Sample size (%)	100	100	0.66
True collocations	2,557	2,293	2,263
Baseline precision (%)	21.02	22.88	22.66

a much larger data set on the effectiveness of the extraction process, we create the *CNC-Surf* reference data set consisting of instances of *PDT-Surf* appearing in the set of 242 million words from *Czech National Corpus* (CNC 2005), the SYN 2000 and SYN 2005 corpora. This data lacks manual annotation but are automatically processed by a POS tagger (Hajič 2004). The collocation candidates are identified as *surface bigrams* in the same fashion as in *PDT-Surf*. The reference data itself contains 9,868 surface bigrams (from the total of 1,503,072 surface bigrams obtained from CNC after applying the frequency and POS filters), the remaining 153 do not occur in it more than five times and were not included in the *CNC-Surf* data set.

2.1 Manual annotation

The reference sets of collocation candidates were manually processed by three trained linguists with the aim of identifying collocations according the notion specified in Sect. 1. It requires collocations to be grammatical units (words in a syntactic relation) that are not entirely predictable (semantically and syntactically). Essentially, the annotators had to decide whether each candidate should be listed in a lexicon or it is a free word combination (only grammatically constrained).

The dependency bigrams from *PDT-Dep* were assessed first. The annotation was performed independently, in parallel, and without knowledge of context. To minimize the cost of the process, each collocation candidate was presented to each annotator only once – although it could appear in various different contexts, which corresponds with the goal of extracting collocations as types not as tokens (instances). The annotators were instructed to judge any bigram which could *eventually* appear in a context where it has a character of collocation as *true collocation*. For example, idiomatic expressions were judged as collocations although they can also occur in contexts where they have a literal meaning. As a result, the annotators were relatively liberal in their judgments, but their full agreement was required to mark a candidate as true collocation in the reference data set. During the assessment, the annotators also attempted to distinguish between

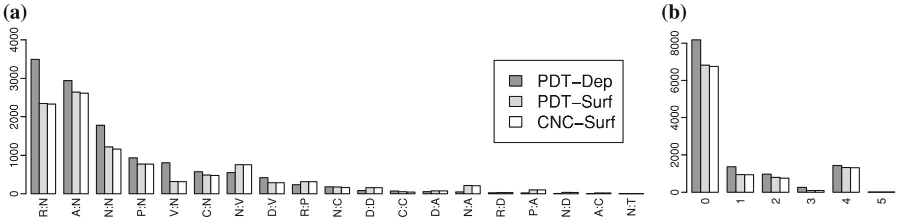

Fig. 1 **a** Part-of-speech pattern distribution in the reference data sets; **b** distribution of collocation categories in the reference data sets assigned by one of the annotators

subtypes of collocations and classified each collocation into one of the categories listed below. This classification, however, was not intended as a result of the annotation process (our primary goal is binary classification) but rather as a way to clarify and simplify the annotation. Any bigram that can be assigned to these categories is considered a true collocation.

1. Stock phrases
   zásadní problém (major problem), konec roku (end of the year)
2. Names of persons, organizations, geographical locations, and other entities
   Pražský hrad (Prague Castle), Červený kříž (Red Cross)
3. Support verb constructions
   mít pravdu (to be right), činit rozhodnutí (make decision)
4. Technical terms
   předseda vlády (prime minister), očitý svědek (eye witness)
5. Idiomatic expressions
   studená válka (cold war), visí otazník (hanging question mark ~ open question)

The surface bigrams from *PDT-Surf* were annotated in the same fashion – but only those collocation candidates that do not appear in *PDT-Dep* were actually judged (974 items). Technically, we removed the syntactic information from the *PDT-Dep* candidates and transfered the annotation to those in *PDT-Surf*. If a surface bigram from *PDT-Surf* appears also in *PDT-Dep* (syntactic relation ignored), it is assigned the same annotation. Similarly, the annotation of *CNC-Surf* was transfered from *PDT-Surf* (the *CNC-Surf* candidates is a subset of the *PDT-Surf* candidates).

The inter-annotator agreement was evaluated on all the candidates from *PDT-Dep* and all the categories of collocations (plus a 0 category for non-collocations) using the Fleiss' κ statistics.[1] Its exact value among all the three annotators was relatively low 0.49. This demonstrates that the notion of collocation is very subjective, domain-specific, and also somewhat vague. In our experiments,

[1] An agreement measure for any numbers of annotators (Fleiss 1971): $\kappa = \frac{P_o - P_e}{1 - P_e}$, where P_o is the relative *observed* agreement among annotators and P_e is the theoretical probability of *chance* agreement (each annotator randomly choosing each category). The factor $1 - P_e$ then corresponds to the level of agreement achievable above chance and $P_o - P_e$ is the level of agreement actually achieved above chance. For two annotators the exact Fleiss' κ reduces to the well known Cohen's κ (Conger 1980).

we do not distinguish between different collocation categories – ignoring them (considering only two categories: *true collocations* and *false collocations*) increased Fleiss' κ among all three annotators to 0.56. The multiple annotation was performed in order to get a more precise and objective idea about what can be considered a collocation by combining independent outcomes of the annotators. Only those candidates that *all* three annotators recognized as collocations (of any type) were considered *true collocations* (full agreement required). The *PDT-Dep* reference data set contains 2,557 such bigrams (21.02% of all the candidates), *PDT-Surf* data set 2,293 (22.88%), and *CNC-Surf* data set 2,263 (22.66%). See Table 1 and Fig. 1 for details.

For all experiments, the data were split into seven stratified subsets each containing the same ratio of collocations. Six folds are intended to be used for six-fold *cross validation* and average performance estimation. The remaining fold is put aside to be used as *held-out* data in further experiments.

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 are supposed to indicate the potential for a candidate to be a collocation. They can be used either 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 expected 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 or surface bigrams). Joint and marginal occurrence 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 (see Table 3 in Appendix, rows 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, see the example in Fig. 2), 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 a specified context window, see the example in Fig. 2), the association measures rank collocations according to the assumption that semantically non-compositional expressions typically occur as (semantic) units in

.. součástí trhu, vznikl **obratem**	**černý trh**	s plyšovými medvídky a ...
zabránit přísunu drog na **domácí**	**černý trh**	v hodnotě 32 milionu .....
stejnými jednotlivci i **kompletní**	**černý trh**	. Jinými slovy, byla by ....
...pomáhali pašování cigaret **na**	**černý trh**	do východního Německa ...
....nájemních práv **nezaručený**	**černý trh**	. Libor Dellin, člen ....

...miliónů dolarů. **Ovlivňuje nějak negativně tento**	**černý trh**	**naše hospodářství?** Je to pouze ztráta na daních
.. **Maltské liry lze nakoupit pouze ve směnárnách,**	**černý trh**	**s valutami neexistuje.** Na Maltě je v porovnání s
operoval i ženu. **A přece má, jak se říká na Arbatu,**	**černý trh**	**něco do sebe.** Je - li hlad nejlepší kuchař, je .....
.. přestal. **V patách za krizí vstoupil do Bělehradu**	**černý trh**	**, pašování a zvýšená kriminalita.** Překupníci ...
........z toho obviněni. **ídí gangy, které kontrolují**	**černý trh**	**a okrádají cizince.** Oba byli zbaveni funkcí a byl

Fig. 2 Example of a *left* immediate context (*top*) and empirical context (*bottom*) of the expression *černý trh* (*black market*). The contexts consist of *non-underlined* words in *bold*

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

The task of collocation extraction can be viewed as binary classification. 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 is usually measured e.g. by *accuracy* – proportion of correct predictions. However, the proportion of the two classes in our case (collocations and non-collocations) is far from equal and we want to distinguish classifier performance between them. In this case, several authors, e.g. Evert and Krenn (2001), suggest using *precision* – proportion of positive predictions correct and *recall* – proportion of positives correctly predicted. The higher the scores the better the classification is.

3.2 Precision-recall curves

Since choosing the 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* (PR) 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.

From the statistical point of view, the precision-recall curves must be viewed as estimates of their true (unknown) shapes from a (random) data sample. As such, they have a certain statistical variance and are sensitive to data. For illustration, see Fig. 3a showing PR curves obtained on each of the six crossvalidation folds of *PDT-Dep* (each of the thin curves corresponds to one data fold). In order to obtain a good estimation of their true shape we must apply some kind of *curve averaging*

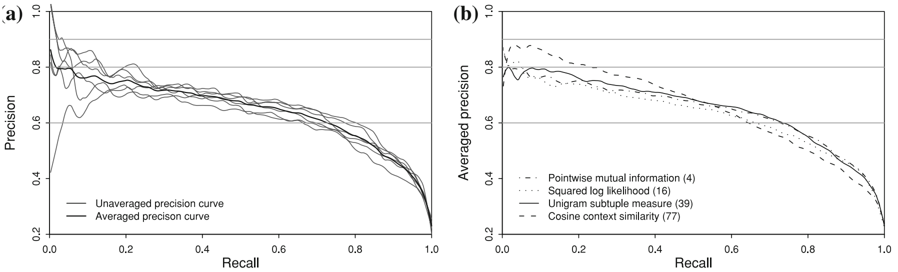

Fig. 3 **a** An example of vertical averaging of precision-recall curves. Thin curves represent individual non-averaged curves obtained by *Pointwise mutual information* (4) on six data folds. **b** Crossvalidated and averaged precision-recall curves of selected association measures (the *numbers* in brackets refer to the table in Appendix)

where all crossvalidation folds with precision-recall scores are combined and a single curve is drawn. Such averaging can be done in three ways (Fawcett 2003): *vertically* – fixing recall, averaging precision, *horizontally* – fixing precision, averaging recall, and *combined* – fixing threshold, averaging both precision and recall (Fawcett 2003). Vertical averaging, as illustrated in Fig. 3a, works reasonably well in our case and is used in our experiments. Thin curves are produced by one association measure on six separate data folds; the thick one is obtained by vertical averaging.

3.3 Mean average precision

Visual comparison of precision-recall curves is a powerful 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 is evidently better (although it might not be significantly 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 of recall) 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 the more appropriate nonparametric *paired Wilcoxon signed-ranked test*.

Due to the unreliable precision scores for low recall and their fast changes for high recall (for illustration see Fig. 3a), we suggest the estimation of AP to be limited only to some narrower interval of recall, e.g. $\langle 0.1, 0.9 \rangle$

3.4 Experiments

Following the scenarios described in the previous section, we perform the following experiment on each of the three data sets. For all collocation candidates, we extract their frequency characteristics (the observed contingency tables) and context

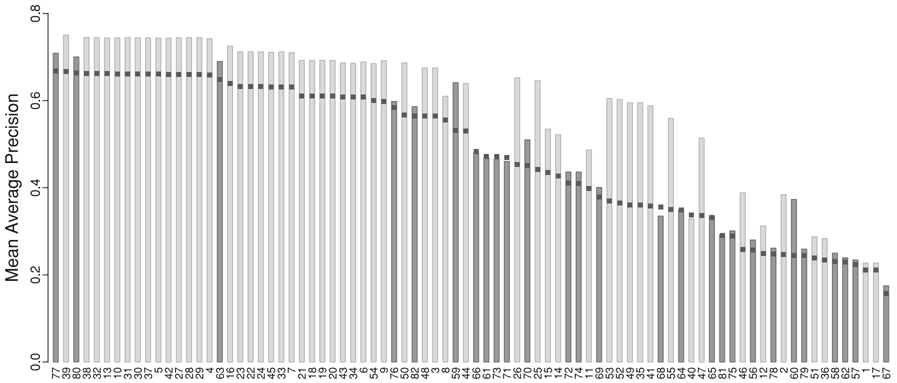

Fig. 4 MAP scores of association measures obtained on *PDT-Surf* (*bars*) and sorted by the descending MAP scores on *PDT-Dep* (*square points*). Methods are referred by numbers from the table in Appendix. The *darker bars* correspond to the context based association measures (56–82)

information (the immediate and empirical contexts) from their source corpora. The empirical contexts are limited to a context window of 3 sentences (the actual one, the preceeding one, and the following one) and filtered to include only open-class word types (noun, adjectives, verbs, adverbs). Based on this information, we compute the scores for all 82 association measures for all the candidates in each evaluation data fold. Then, for each association measure and each fold, we rank the candidates according to their descending association scores, compute values of precision and recall after each true collocation appearing in the ranked list, plot the averaged precision-recall curve, and compute the average precision on the recall interval $\langle 0.1, 0.9 \rangle$. The AP values obtained on the evaluation data folds are used to estimate the mean average precision as the main evaluation measure. Further, we rank the association measures according to their MAP values in descending order and depict the results in a graph. Finally, we apply the paired Wilcoxon test and identify association measures with statistically indistinguishable performance.

First, we evaluate the association measures on *PDT-Dep*, the set of dependency bigrams extracted from *Prague Dependency Treebank*. A baseline system ranking the *PDT-Dep* candidates randomly would operate with the expected precision (and also MAP) of 21.02%, which is the prior probability of a collocation candidate to be a true collocation. Precision-recall curves of some well-performing methods are plotted in Fig. 3b. The best method evaluated by mean average precision is *Cosine context similarity in boolean vector space (77)* with MAP = 66.79%, followed by *Unigram subtuple measure (39)*, MAP = 66.72% and other 14 association measures with nearly identical and statistically indistinguishable performance (see the dark square points in Fig. 4). They include some popular methods known to perform reliably in this task, such as *Pointwise mutual information (4), Mutual dependency (5), Pearson's χ^2 test (10), Z score (13)*, or *Odds ratio (27)*. Surprisingly, another commonly used method *T test (12)* only achieves MAP = 24.89% and performes slightly above the baseline. Although the best association measure uses the empirical context information, most of the other context-based methods are concentrated in the second half of the ranked list of the measures (indicated by dark-gray bars) and do not perform well.

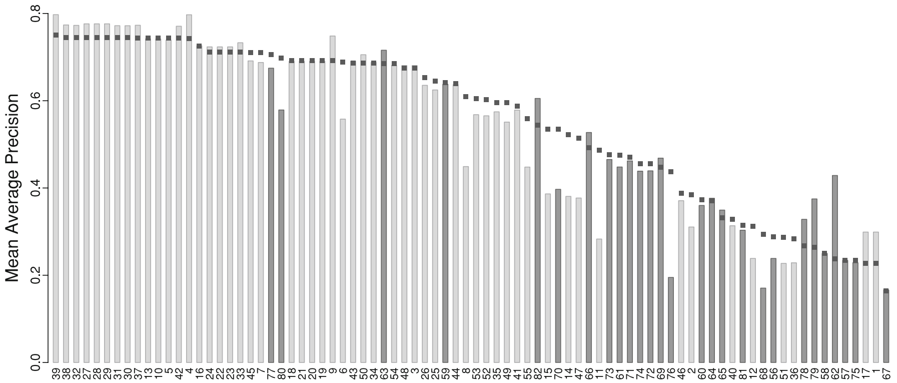

Fig. 5 MAP scores of association measures computed on *CNC-Surf* (*bars*) and sorted by the descending scores of MAP on *PDT-Surf* (*square points*). Methods are referred by numbers from the table in Appendix

As a second experiment, we perform a similar procedure on the same text from the *Prague Dependency Treebank* (exploiting only the morphological information), compute association scores for surface bigrams from the *PDT-Surf* data set and also show them in Fig. 4. For a better comparison, the methods are sorted according to the results obtained on *PDT-Dep*. The MAP scores of most association measures increased dramatically in this experiment. The best performing method is *Unigram subtuple measure (39)* with MAP = 75.03% compared to 66.71% achieved on the dependency bigrams (absolute improvement of 11.68%). This is probably due to the non-directly-adjacent dependency bigrams not appearing in the *PDT-Surf* data set: in most cases, they do not form collocations. Interestingly, this improvement is not so significant for context-based association measures (see the dark-gray bars in Fig. 4). The best context-based measure on *PDT-Dep* (77) ended up as the 22nd on the surface data and its score increased only by absolute 4.1%

The third experiment is performed analogously on the the *CNC-Surf* reference data set, i.e. instances of *PDT-Surf* in the *Czech National Corpus*. The content of these two data sets is almost the same, *CNC-Surf* shares 98.46% of the collocation candidates with *PDT-Surf*. The main difference is in their frequency counts obtained from their source corpora. The data from the *Czech National Corpus* are approximately 150 times larger (in terms of the number of tokens). The average frequency of candidates in *PDT-Surf* is 161 compared to 1,662 in *CNC-Surf*.

The results are presented in Fig. 5 and compared to those obtained on *PDT-Surf*. The effect of using a much larger data set leading to better occurrence probability estimations is positive only for certain methods – surprisingly the most effective ones. A significant improvement (4.5 absolute percentage points on average) is observed only for a few of the best performing association measures on *PDT-Surf* and also for some other less efficient methods. Performance of other association measures does not significantly change or it drops down. The two most appropriate measures are *Unigram subtuple measure (39)* with MAP = 79.74% and *Pointwise mutual information (4)* with MAP = 79.71%, known to be very effective on large data.

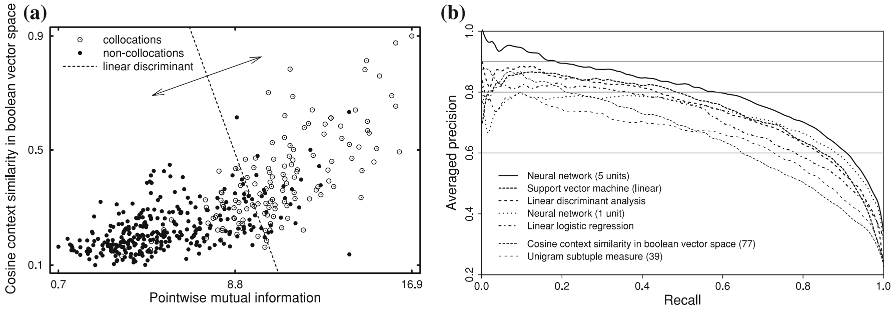

Fig. 6 **a** Visualization of scores of two association measures. The *dashed line* denotes a linear discriminant obtained by logistic linear regression. By moving this boundary we can tune the classifier output (a 5% stratified sample of the *PDT-Dep* data set is displayed). **b** Precision-recall curves of selected methods combining all association measures compared with curves of two best measures employed individually on the same data sets

When comparing results on these data sets, we must be aware of the fact that the baseline MAP scores on these data sets are not equal (21.02% for *PDT-Dep*, 22.88% for *PDT-Surf*, 22.66% for *CNC-Surf*) and their differences must be taken into account during the analysis of the MAP scores on different data sets. However, these differences are relatively small compared to the differences in MAP of association measures observed in our experiments.

An interesting point to note is that the context similarity measures on the *PDT-Dep* data set, e.g. (77) slightly outperform measures based on simple occurrence frequencies, e.g. (39), measured by MAP. A more thorough comparison by precision-recall curves shows that the former very significantly predominates the latter in the first half of the recall interval and vice versa in the second half (Fig. 3b). This is a case where MAP is not a sufficient metric for comparing performance of association measures. 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* measures will be essential in the following sections.

4 Combining association measures

A motivating example for combining association measures is shown in Fig. 6: association scores of *Pointwise mutual information* and *Cosine context similarity* are independent enough to be linearly combined in one model and to achieve better results.

Each collocation candidate x^i can be described by the *feature vector* $\mathbf{x}^i = (x_1^i, \ldots, x_{82}^i)^T$ consisting of all 82 association scores from the table in Appendix 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 $f(\mathbf{x}) \rightarrow \mathbb{R}$ that would determine the strength of lexical association between components of bigram x and hence have 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 and how to optimize its parameters. 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:

$$\operatorname{logit}(\pi) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$$

where $\operatorname{logit}(\pi) = \log(\pi/(1-\pi))$ is a canonical link function for modeling binary response 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. After projection, $\mathbf{c}\ ^T\mathbf{x}$ can be directly used as ranker.

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

4.3 Support vector machines

For technical reason, we now change the labels from $y^i \in \{0,\ 1\}$ to $y^i \in \{-1,+1\}$. 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 $y(\mathbf{x}) = \operatorname{sign}(f(\mathbf{x}))$ 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* $L(y, f(\mathbf{x})) = [1 - y\ f(\mathbf{x})]^+$ 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 the 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 the often inappropriate grid search, we employ the effective algorithm which fits the entire SVM regularization path $[\beta_0(\lambda), \boldsymbol{\beta}(\lambda)]$ and gives us the option to choose the optimal value of λ proposed by Hastie et al. (2004). The total loss on training data is used as the objective function.

4.4 Neural networks

Using 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

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

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 the 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 likelihood*. To avoid problems with convergence of the algorithm we used the former one. As the tuning parameter of a classifier, the number of units in the hidden layer is used.

The presented methods are originally intended for (binary) classification. For our purposes, they are used with the following modification: In the training phase, they are employed as regular classifiers on two-class training data (collocations and non-collocations) to fit the model parameters. In the application phase, no classification threshold applies and for each collocation candidate the ranker function computes a value which is interpreted as the association score. Applying the classification threshold would turn the ranker back into a regular classifier. The candidates with higher scores would fall into one class (collocations), the rest into the other class (non-collocations).

4.5 Experiments

To address the incommensurability of association measures in our experiments, we use a common preprocessing technique for multivariate *standardization*: the values of each association measure are centered towards zero and scaled to unit variance. Precision-recall curves of all methods are obtained by vertical averaging in six-fold cross validation on the same reference data sets as in the earlier experiments. Mean average precision is computed from average precision values estimated on the recall interval $\langle 0.1,0.9\rangle$. In each cross-validation step, five folds are used for training and one fold for testing.

First, we study the performance of the combination methods on the *PDT-Dep* reference data set. All combination methods work very well and gain a substantial performance improvement in comparison with individual measures. The best result is achieved by the neural network with five units in the hidden layer (NNet.5) with MAP = 80.93 %, which is 21.17% relative and 14.08% absolute improvement compared to the best individual association measure. More detailed results are given in Table 2 and corresponding precision-recall curves are depicted in Fig. 6b. We observe a relatively stable improvement within the whole interval of recall.

The neural network is the only method which performs better in its more complex variant (with up to five units in the hidden layer). More complex models, such as neural networks with more than five units in the hidden layer, support vector

Table 2 Performance of methods combining all association measures on *PDT-Dep*: averaged precision (in %) at fixed points of recal and mean average precision (MAP) on the recall interval ⟨0.1,0.9 ⟩ and its relative improvement (+, in %)

Method	Averaged precision at			MAP	
	$R = 20$	$R = 50$	$R = 80$	$R = \langle 0.1,0.9 \rangle$	+
Neural network (5 units)	91.00	81.75	70.22	80.87	21.08
Linear logistic regression	86.96	79.74	64.63	77.36	15.82
Linear discriminant analysis	85.99	77.34	61.44	75.16	12.54
Neural network (1 unit)	82.47	77.08	65.75	74.88	12.11
Support vector machine (linear)	81.33	76.08	61.49	73.03	9.35
Cosine similarity (77)	80.88	68.46	49.99	66.79	0.00
Unigram subtuples (39)	75.86	68.19	55.13	66.72	–

machines with higher order polynomial kernels, quadratic logistic regression, or quadratic discriminant analysis, overfit the training data folds, and better scores are achieved by their simpler variants.

Comparison of performance of all the combination methods on all the reference data sets is presented in Fig. 7a. We observe the same effect as with the individual association measures. Extracting collocations as surface bigrams from PDT with a neural network (5 units in the hidden layer) increases MAP from 80.87% to 84.84% (3.97% absolute improvement). Using the large data from the *Czech National Corpus* (providing much better occurrence probability estimations) adds other 1.46 absolute percentage points and the best MAP score on the *CNC-Surf* reference data increases to 86.30%. This number can be considered as the estimation of MAP (on the recall interval ⟨0.1,0.9 ⟩) that can be achieved with the neural network using all lexical association measures on the entire candidate data extracted from the *Czech National Corpus* and filtered by the part-of-speech and frequency filter (1.5 million surface bigrams), which is a quite promising result.

Our next experiment is focused on the learning process of the employed classification methods. Figure 7b visualizes the *learning curve* of the best performing method (NNet.5) on the *PDT-Dep* data set, i.e. to what extent its performance depends on the size of the training data. The beginning of the curve is fairly steep and we reach 90% of its maximum value with only 5% of the training data, with 15% of the training data we climb to 95%. A system operating with 99% of the maximum MAP score can be developed with 60% of the training data.

5 Model reduction

We have demonstrated that combining association measures in general is reasonable and helps in the collocation extraction task. However, the combination models presented in the previous section are too complex in number of predictors used: some association measures are very similar (analytically or empirically) and in combination hence redundant. They make training of the model difficult and should

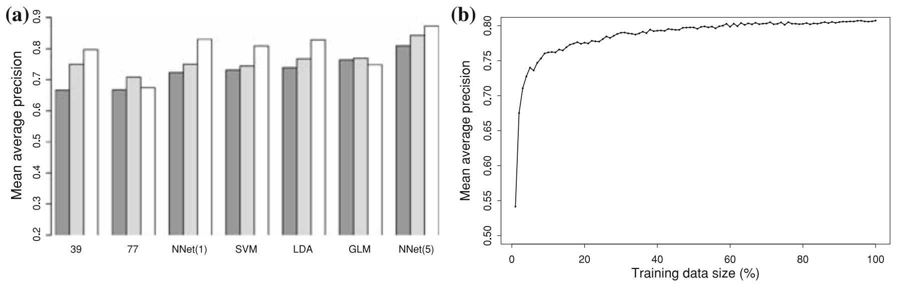

Fig. 7 **a** Performance of methods combining all association measures obtained from the three reference data sets: *PDT-Dep* (*dark gray*), *PDT-Surf* (*gray*), *CNC-Surf* (*white*). **b** The learning curve of the neural network (5 units) measured on the *PDT-Dep* reference data set

be avoided. Some other measures are for this task simply improper, they can hurt the model's performance and should be avoided too.

Experiments with *principal component analysis* applied to the association scores of collocation candidates from the *PDT-Dep* data set show that 95% of its total variance is explained by only 17 principal components and 99.9% is explained by 42 components. Based on this observation, we can expect the number of variables in our models can be significantly reduced with very limited performance degradation.

In this section, we propose an algorithm which eliminates the model variables (association measures) based on two criteria: (1) their linear correlation with other variables in the model and (2) poor contribution to efficient ranking of collocation candidates.

First, we employ *hierarchical clustering* in order to group highly correlated measures into clusters. This clustering is based on the similarity matrix formed by the absolute values of *Pearson's correlation coefficient* computed for each pair of association measures estimated from the held-out data fold (which is independent from the evaluation data folds). This technique starts with each variable in a separate cluster and merges them into consecutively larger clusters based on the values from the similarity matrix until a desired number of clusters is reached or the similarity between clusters exeeds a limit. An example of a complete hierarchical clustering of association measures is depicted in Fig. 8. If the stopping criterion is set properly, the measures in each cluster have an approximately equal contribution to the model. Only one of them is selected as representative and used in the reduced model (the other measures are redundant). The selection can be random or based e.g. on the (absolute) individual performance of the measures on the held-out data fold.

The reduced model at this point does not contain highly-correlated variables and can be more easily fit (trained) to the data. However, these variables are not guaranteed to have a positive contribution to the model. Therefore, the algorithm continues with the second step and applies a standard *step-wise* procedure removing one variable in each iteration, causing minimal degradation of the model's performance measured by MAP on the held-out data fold. The procedure stops when the degradation becomes statistically significant by the paired Wilcoxon signed-rank test.

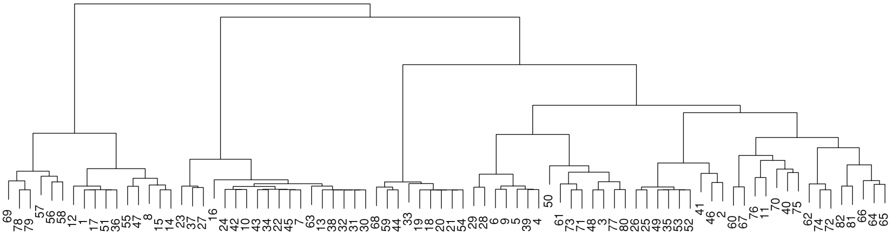

Fig. 8 A dendrogram – visualization of hierarchical clustering on the held-out data of the *PDT-Dep* data set

5.1 Experiments

We test the model reduction experiment on the neural network model with five units in the hidden layer (the best performing combination method) on the *PDT-Dep* reference data set. The parameter (number of clusters) is experimentally set to 60. In each iteration of the algorithm, we use five data folds (out of the six 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 is cross-validated on the same six data-folds as in the previous experiments.

Precision-recall curves for some intermediate models are shown in Fig. 9. We can conclude that we are able to reduce the NNet model to 13 predictors without statistically significant difference in performance ($\alpha = 0.05\%$). The corresponding association measures are marked in Table 3 in Appendix. The step-wise phase of the model-reduction is, however, very sensitive to data and can easily lead to different results.

6 Conclusions

In this work we have attempted to evaluate lexical association measures employed for automatic collocation extraction.

We have created and manually annotated three reference data sets for three evaluation scenarios: extracting collocations as *dependency bigrams* from the morphologically and syntactically annotated *Prague Dependency Treebank* (*PDT-Dep*), extracting collocations from the same source as *surface bigrams* (*PDT-Surf*), and extracting collocations as *surface bigrams* from the *Czech National Corpus* with automatically assigned morphological tags (*CNC-Surf*). The data sets contain 9–12 thousand collocation candidates and were manually processed by three linguists in parallel. About 20% of the bigrams in each data set were agreed to be collocations by three annotators and considered *true collocations* for the evaluation.

We have implemented 82 association measures and evaluated them against the three reference data set by averaged *precision-recall curves* and *mean average precision* in six-fold cross validation. The best result on *PDT-Dep* has been achieved by a method measuring *Cosine context similarity in boolean vector space* with mean average precision of 66.79%. Extracting collocations as surface bigrams

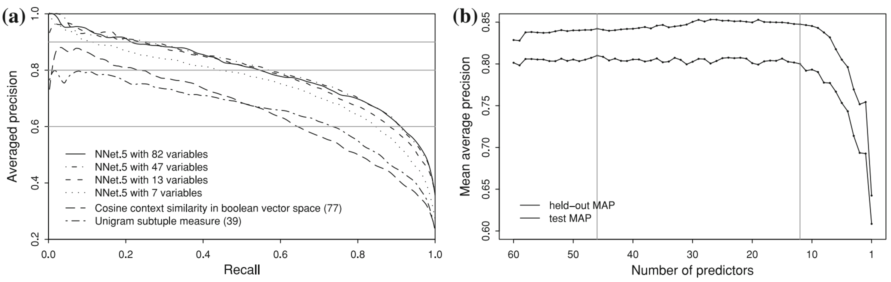

Fig. 9 **a** Precision-recall curves of reduced neural network models compared with curves of the full model and two best individual methods. **b** MAP scores from the interation of the model reduction process applied on the neural network (5 units)

have been shown to be also effective approach. The results of almost all measures obtained on *PDT-Surf* have been significantly improved: the best MAP of 75.03% has been achieved by the *Unigram subtuple measure*. The experiments carried out on *CNC-Surf* have shown that processing of a larger corpus has a positive effect on the quality of collocation extraction; MAP scores of the *Unigram subtuple measure* and *Pointwise mutual information* have increased up to 79.7%.

Furthermore, we have evaluated four classification methods combining multiple association measures and demonstrated that this approach certainly helps in the collocation extraction task. All investigated methods have greatly outperformed individual association measures on all reference data sets. The best results have been achieved by a simple neural network with five units in the hidden layer. Its mean average precision of 80.87% achieved on *PDT-Dep* have represents 21.08% relative improvement with respect to the best individual measure. In the experiments on *CNC-Surf* we have estimated the expected value of MAP on the entire candidate data as 86.30%. The learning curve of the neural network model on the *PDT-Dep* data set demonstrates that the amount of training data used in our experiments is not necessary. We can develop a system with only 15% of the training data and achieve 95% of MAP of the model trained on all data. By the proposed model reduction procedure we are also able to reduce the number of variables in the neural network from 82 to 13 without significant degradation of its performance.

In our work, we have not attempted 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 the notion of collocation itself. Instead, we have demonstrated that combining association measures is meaningful and improves precision and recall of the extraction procedure and the full performance improvement can be achieved by a relatively small number of measures combined.

Acknowledgments This is a revised and extended version of our previous work (Pecina and Schlesinger 2006). Details on the reference data sets are described in (Pecina 2008a). Experiments that are performed on other data sets and confirm good results of our combination methods are presented in (Pecina 2008b). This work was supported by the Ministry of Education of the Czech Republic project MSM 0021620838.

Appendix

Table 3 The inventory of lexical association measures used for collocation extraction used in our experiments

#	*Name*	*Formula*
1.	**Joint probability**	$P(xy)$
2.	**Conditional probability**	$P(y\|x)$
3.	**Reverse conditional probability**	$P(x\|y)$
4.	**Pointwise mutual information**	$\log \frac{P(xy)}{P(x*)P(*y)}$
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**	$\sum_{i,j} \frac{(f_{ij}-\hat{f}_{ij})^2}{\hat{f}_{ij}}$
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**	$\frac{\hat{f}(xy)-f(xy)\log \hat{f}(xy)+\log f(xy)!}{\log N}$
15.	**Log likelihood ratio**	$-2\sum_{i,j} f_{ij} \log \frac{f_{ij}}{\hat{f}_{ij}}$
16.	**Squared log likelihood ratio**	$-2\sum_{i,j} \frac{\log f_{ij}^2}{\hat{f}_{ij}}$
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**	$\frac{1}{2}\left(\frac{a}{a+b} + \frac{a}{a+c}\right)$
26.	**Fourth Sokal-Sneath**	$\frac{1}{4}\left(\frac{a}{a+b} + \frac{a}{a+c} + \frac{d}{d+b} + \frac{d}{d+c}\right)$
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**	$\frac{ad}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$

Table 3 continued

#	*Name*	*Formula*
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**	$\max[P(xy)\log\frac{P(y\|x)}{P(*y)}+P(x\bar{y})\log\frac{P(\bar{y}\|x)}{P(*\bar{y})},$ $P(xy)\log\frac{P(x\|y)}{P(x*)}+P(\bar{x}y)\log\frac{P(\bar{x}\|y)}{P(\bar{x}*)}]$
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**	$\frac{P(xy)+P(\bar{x}\bar{y})}{P(x*)P(y)+P(\bar{x}*)P(*y)}\cdot$ $\frac{1-P(x*)P(*y)-P(\bar{x}*)P(*y)}{1-P(xy)-P(\bar{x}\bar{y})}$
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**	$P(*y)\log P(*y)-\sum_w P(w\|C^r_{xy})\log P(w\|C^r_{xy})$

Table 3 continued

#	*Name*	*Formula*
61.	**Cross entropy**	$-\sum_w P(w \mid C_x)\log P(w \mid C_y)$
$\star$62.	**Reverse cross entropy**	$-\sum_w P(w \mid C_y)\log P(w \mid C_x)$
63.	**Intersection measure**	$\frac{2\lvert C_x \cap C_y\rvert}{\lvert C_x\rvert+\lvert C_y\rvert}$
64.	**Euclidean norm**	$\sqrt{\sum_w (P(w \mid C_x)-P(w \mid C_y))^2}$
65.	**Cosine norm**	$\frac{\sum_w P(w \mid C_x)P(w \mid C_y)}{\sum_w P(w \mid C_x)^2 \cdot \sum_w P(w \mid C_y)^2}$
66.	***L1* norm**	$\sum_w \lvert P(w \mid C_x)-P(w \mid C_y)\rvert$
67.	**Confusion probability**	$\sum_w \frac{P(x \mid C_w)P(y \mid C_w)P(w)}{P(x*)}$
$\star$68.	**Reverse confusion probability**	$\sum_w \frac{P(y \mid C_w)P(x \mid C_w)P(w)}{P(*y)}$
69.	**Jensen-Shannon divergence**	$\frac{1}{2}[D(p(w \mid C_x)\Vert\frac{1}{2}(p(w \mid C_x)+p(w \mid C_y)))$
		$+D(p(w \mid C_y)\Vert\frac{1}{2}(p(w \mid C_x)+p(w \mid C_y)))]$
70.	**Cosine of pointfwise *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**	$\sum_w P(w \mid C_x)\log\frac{P(w \mid C_x)}{P(w \mid C_y)}$
72.	**Reverse KL divergence**	$\sum_w P(w \mid C_y)\log\frac{P(w \mid C_y)}{P(w \mid C_x)}$
73.	**Skew divergence**	$D(p(w \mid C_x)\Vert\alpha p(w \mid C_y)+(1-\alpha)p(w \mid C_x))$
74.	**Reverse skew divergence**	$D(p(w \mid C_y)\Vert\alpha p(w \mid C_x)+(1-\alpha)p(w \mid C_y))$
$\star$75.	**Phrase word coocurrence**	$\frac{1}{2}(\frac{f(x \mid C_{xy})}{f(xy)}+\frac{f(y \mid C_{xy})}{f(xy)})$
76.	**Word association**	$\frac{1}{2}(\frac{f(x \mid C_y)-f(xy)}{f(xy)}+\frac{f(y \mid C_x)-f(xy)}{f(xy)})$
Cosine context similarity:		$\frac{1}{2}(\cos(\mathbf{c}_x,\mathbf{c}_{xy})+\cos(\mathbf{c}_y,\mathbf{c}_{xy}))$
		$\mathbf{c}_z=(z_i);\ \cos(\mathbf{c}_x,\mathbf{c}_y)=\frac{\sum x_i y_i}{\sqrt{\sum {x_i}^2}\cdot\sqrt{\sum {y_i}^2}}$
$\star$77.	**in boolean vector space**	$z_i=\delta(f(w_i \mid C_z))$
78.	**in tf vector space**	$z_i=f(w_i \mid C_z)$
79.	**in $tf \cdot idf$ vector space**	$z_i=f(w_i \mid C_z)\cdot\frac{N}{df(w_i)};\ df(w_i)=\lvert\{x : w_i \varepsilon C_x\}\rvert$
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**	$z_i=\delta(f(w_i \mid C_z))$
$\star$81.	**in tf vector space**	$z_i=f(w_i \mid C_z)$
$\star$82.	**in $tf \cdot idf$ vector space**	$z_i=f(w_i \mid C_z)\cdot\frac{N}{df(w_i)};\ df(w_i)=\lvert\{x : w_i \varepsilon C_x\}\rvert$

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

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

A contingency table contains observed joint 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 $\hat{f}(xy)\ = f(x*)f(*y)/N$

References

Bartsch, S. (2004). *Structural und functional properties of collocations in English. A corpus study of lexical and pragmatic constraints on lexical co-occurrence*. Tübingen: Gunter Narr Verlag.

Berry-Rogghe, G. L. (1973). The computation of collocations and their relevance in lexical studies. In *The computer and literal studies* (pp. 103–112). Edinburgh, New York: University Press.

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

Choueka, Y., Klein, S., & Neuwitz, E. (1983). Automatic retrieval of frequent idiomatic and collocational expressions in a large corpus. *Journal of the Association for Literary and Linguistic Computing, 4*(1), 34–38.

Church, K., & Hanks, P. (1990). Word association norms, mutual information and lexicography. *Computational Linguistics, 16*(1), 22–29.

Conger, A. J. (1980). Integration and generalisation of Kappas for multiple raters. *Psychological Bulletin, 88*, 322–328

Daille, B. (1996). Study and implementation of combined techniques for automatic extraction of terminology. In J. L. Klavans & P. Resnik (Eds.), *The balancing act* (Chap. 3, pp. 49–66). Cambridge, MA: MIT Press.

Evert, S. (2004). The statistics of word cooccurrences: Word pairs and collocations. PhD Thesis, University of Stuttgart.

Evert, S., & Krenn, B. (2001). Methods for the qualitative evaluation of lexical association measures. In *Proceedings of the 39th annual meeting of the association for computational linguistics* (pp. 188–195).

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

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin 76*, 378–382.

Hajič, J. (2004). *Disambiguation of rich inflection (computational morphology of Czech)* (Vol. 1). Prague: Charles University Press.

Hastie, T., Rosset, S., Tibshirani, R., & Zhu, J. (2004). The entire regularization path for the support vector machine. *Journal of Machine Learning Research, 5*, 1391–1415.

Hull, D. (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.

Inkpen, D., & Hirst, G. (2002). Acquiring collocations for lexical choice between near synonyms. In *SIGLEX workshop on unsupervised lexical acquisition, 40th meeting of the ACL*, Philadelphia.

Kita, K., Kato, Y., Omoto, T., & Yano, Y. (1994). A comparative study of automatic extraction of collocations from corpora: Mutual information vs. cost criteria. *Journal of Natural Language Processing, 1*(1), 21–33.

Krenn, B. (2000). The usual suspects: Data-oriented models for identification and representation of lexical collocations. PhD Thesis, Saarland University.

Manning, C. D., & Schütze, H. (1999). *Foundations of statistical natural language processing*. Cambridge, Massachusetts: The MIT Press, Chap. 5. Collocations.

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

Palmer, H. E. (1938). *A grammar of English words*. London: Longman

PDT (2006). Prague dependency treebank 2.0. Institute of Formal and Applied Lingustics.

Pearce, D. (2002) A comparative evaluation of collocation extraction techniques. In *Third international conference on language resources and evaluation*. Spain, Las Palmas.

Pecina, P. (2008a). Machine learning approach to mutliword expression extraction. In *Proceedings of the sixth international conference on language resources and evaluation workshop: Towards a shared task for multiword expressions (MWE 2008)*, Marrakech, Morocco.

Pecina, P. (2008b). Reference data for Czech collocation extraction. In *Proceedings of the sixth international conference on language resources and evaluation workshop: Towards a shared task for multiword expressions (MWE 2008).* Marrakech, Morocco.

Pecina, P., & Schlesinger, P. (2006) Combining association measures for collocation extraction. In *Proceedings of the 21th international conference on computational linguistics and 44th annual meeting of the association for computational linguistics (COLING/ACL 2006).* Sydney, Australia.

Shimohata, S., Sugio, T., Nagata, J. (1997). Retrieving collocations by co-occurrences and word order constraints. In *Proceedings of the 35th meeting of ACL/EACL* (pp. 476–481). Madrid, Spain.

Smadja, F. (1993). Retrieving collocations from text: Xtract. *Computational Linguistics, 19*, 143–177

Stevens, M. E., Giuliano, V. E., & Heilprin, L. B. (Eds.), (1965). *Proceedings of the symposium on statistical association methods for mechanized documentation* (Vol. 269). Washington, DC: National Bureau of Standards Miscellaneous Publication.

Venables, W. N., & Ripley, B. (2002). *Modern applied statistics with S* (4th ed.). New York: Springer.

Zhai, C. (1997). Exploiting context to identify lexical atoms: A statistical view of linguistic context. In *International and interdisciplinary conferences on modeling and using context*.