University of Essex

Natural Language Engineering and Web Applications Group
Depratment of Computer Science

Kappa^3 = Alpha (or Beta)

Ron Artstein
artstein [at] essex.ac.uk

Massimo Poesio
poesio [at] essex.ac.uk

NLE Technical Note 05-1

CS Technical Report CSM-437
ISSN 1744-8050

29 September 2005

Contents

University of Essex Natural Language Engineering and Web Applications Group
NLE Technical Note 05-1, CS Technical Report CSM-437, ISSN 1744-8050
29 September 2005

Kappa^3 = Alpha (or Beta)

Ron Artstein
artstein [at] essex.ac.uk

Massimo Poesio
poesio [at] essex.ac.uk

Abstract This paper (i) clarifies a number of points concerning coefficients of agreement; (ii) revisits the issue of bias discussed by Di Eugenio and Glass, showing that the difference due to bias between κ and π disappears as the number of annotators grows; (iii) fills a few gaps in the literature, e.g., by introducing a new coefficient called β, which generalizes Krippendorff's α to take into account bias; (iv) argues that weighted, α-like coefficients, possibly with bias, are probably more helpful than κ in many, if not all, annotation tasks, but make the problems of interpreting the values of the coefficient and comparing it across tasks even more difficult.

1 Introduction and Motivations

Ever since the mid-Nineties, increasing effort has gone into putting semantics and discourse research on the same corpus-based footing as other areas of Computational Linguistics (CL). This soon led to worries about the subjectivity of the judgments required to create annotated resources for semantics and pragmatics, much greater than for the aspects of language interpretation of concern to the first resource creation efforts. Early attempts to develop new techniques to assess coders' agreement on segmentation tasks (such as Passonneau and Litman 1993) led Carletta (1996) to suggest the adoption of the K coefficient of agreement, a variant of Cohen's κ (Cohen 1960), as this had already been used for similar purposes in content analysis for a long time.[1] Carletta's proposals were enormously influential, and K quickly became the de-facto standard in work on discourse (Carletta et al. 1997; Hearst 1997; Poesio and Vieira 1998; Marcu et al. 1999; Di Eugenio 2000) as well as in other areas of Computational Linguistics (e.g., Bruce and Wiebe 1998; Stevenson and Gaizauskas 2000). However, a number of issues about the suitability of K were also raised, some already in Carletta's own work (Carletta et al. 1997), ranging from simple questions about the way the coefficient is computed (e.g., whether it really applies when more than two coders are used), to debates about which levels of agreement can be considered 'acceptable', to the realization, finally, that K is not appropriate for all types of agreement (Poesio and Vieira 1998; Marcu et al. 1999; Di Eugenio 2000; Stevenson and Gaizauskas 2000).

[1] As we will see below, there are lots of terminological inconsistencies in the literature. Carletta uses the term kappa for the coefficient of agreement, referring to Krippendorff (1980) and Siegel and Castellan (1988) for an introduction, and using Siegel and Castellan's terminology and definitions. However, Siegel and Castellan's statistic, which they call K, is actually Fleiss's generalization to more than two coders of Scott's π, not of the original Cohen's κ; to confuse matters further, Siegel and Castellan use the term κ to indicate the parameter which is estimated by K (i.e., a function of K with an approximately normal distribution which can be used to estimate the significance of the value of K obtained). In what follows, we will use the term κ to indicate coefficients that do not make the marginal homogeneity assumption (the assumption that coders have the same beliefs about the chance distribution of items into categories) – Cohen's original coefficient and its generalization to more than two coders – and use the term K for the coefficient discussed by Siegel and Castellan.

Now that ten years have passed from Carletta's original presentation at the workshop on Empirical Methods in Discourse, an effort to reconsider the use of K is under way, and already at least two important articles have appeared. Di Eugenio and Glass (2004) point out that the original κ developed by Cohen (1960) is based on very different assumptions about coder bias from K of Siegel and Castellan (1988), which is typically used in CL, and that choosing one or the other may lead to different reliability values – e.g., to cross the dreaded .67 threshold. A second development was Passonneau's suggestion (2004) that Krippendorf's α (Krippendorff 1980) is better than kappa for measuring agreement on anaphoric annotation. In this paper, the third re-examination of the issue, we have four main goals. First of all, we want to clarify further the relation between the many kappa-like measures and other correlation coefficients, and between the many distinct coefficients for nominal scales proposed in the literature. The second goal is to address the issue of bias raised by Di Eugenio and Glass: we'll propose a way to measure bias, and prove that bias gets smaller as the number of annotators grows. Third, we introduce a biased variant of Krippendorff's α, called β. Finally, we want to open the discussion on the usefulness of these measures for CL purposes, suggesting that α or β are more appropriate than kappa for many purposes, including segmentation and wordsense tagging.

2 Coefficients of Agreement

2.1 A common notation

The discussions of coefficients of agreement found in the literature often use different notations to express similar concepts. We will introduce a uniform notation, which we hope will make the relations between the various coefficients of agreement clearer. First of all, we will use the following conventions:

- The set of items is $\{ i \mid i \in I \}$ and is of cardinality **i**.
- The set of categories is $\{ k \mid k \in K \}$ and is of cardinality **k**.
- The set of coders is $\{ c \mid c \in C \}$ and is of cardinality **c**.

Confusion also arises from the use of the letter *P*, which is used with at least three distinct interpretations, namely "proportion", "percent", and "probability". We will use the following interpretations uniformly throughout the paper.

- We will use the notation A_o (observed agreement) and D_o (observed disagreement) to indicate the observed agreement and disagreement.
- The notation A_e and D_e will be used to indicate expected agreement and expected disagreement, respectively. The relevant coefficient will be indicated with a superscript when an ambiguity may arise (for example, A_e^π is the expected agreement used for calculating π, and A_e^κ is the expected agreement used for calculating κ).
- The notation $P(\cdot)$ will be reserved for the probability of a variable.

Finally, we will use **n** with a subscript parameter to indicate the number of judgments of a particular type:

- $\mathbf{n}_{ik}$ is the number of coders who assigned item *i* to category *k*;
- $\mathbf{n}_{ck}$ is the number of items assigned by coder *c* to category *k*;
- $\mathbf{n}_k$ is the total number of items assigned by all coders to category *k*.

2.2 Unsatisfactory Measures of Agreement

Why do we need new coefficients to measure agreement? Couldn't we use simpler measures such as percentage agreement, or traditional statistics like χ^2? Although the answer to this question has been given a number of times in the literature, we will go over it again, in part for completeness' sake, but also to clarify the problems that kappa-like measures are meant to solve.

Percentage Agreement The simplest measure of agreement between two coders is **percentage of agreement** or **observed agreement**, defined by Scott (1955, page 323) as "the percentage of judgments on which the two analysts agree when coding the same data independently". This is the number of items on which the coders agree divided by the total number of items. More precisely, and looking ahead to the discussion below, observed agreement is the arithmetic mean of the **agreement value** agr_i for all items $i \in I$, defined as follows:

$$\text{agr}_i = \begin{cases} 1 & \text{if the two coders assign } i \text{ to the same category} \\ 0 & \text{if the two coders assign } i \text{ to different categories} \end{cases}$$

Observed agreement over all the values agr_i for all $i \in I$ is then:

$$A_o = \frac{1}{\mathbf{i}} \sum_{i \in I} \text{agr}_i$$

For example, let us assume we have a very simple annotation scheme for dialogue acts in information-seeking dialogues making a binary distinction between **Statements** and **Info-Requests**, as in the DAMSL dialogue act scheme (Allen and Core 1997), and that our two coders classify 100 utterances according to this scheme as shown in Table 1. Then percentage agreement for this experiment is obtained by summing up the cells on the diagonal and dividing by the total number of items: $A_o = (20 + 50)/100 = .7$

		Coder A		
		Stat	IReq	Total
Coder B	Stat	20 (.2)	20 (.2)	40 (.4)
	IReq	10 (.1)	50 (.5)	60 (.6)
	Total	30 (.3)	70 (.7)	100 (1)

Table 1: A simple example of agreement on dialogue act tagging

Observed agreement is a component in all measures of agreement we consider, but when used on its own it does not yield values that can be compared across studies, as it doesn't tell us how much of the agreement we observe is due to chance. This causes two problems. First of all, as Scott (1955, page 322) points out, "[percentage agreement] is biased in favor of dimensions with a small number of categories": in other words, given two coding schemes for the same phenomenon, the one with fewer categories will result in higher percentage agreement just by chance. Suppose we want to refine the simple binary coding scheme just discussed by introducing a new category of **Checks**, as in the MapTask coding scheme (Carletta et al. 1997). Then if two coders randomly classify utterances in a uniform manner using the first scheme, we would expect them to agree on the classification of half of the items ($\frac{1}{2} \times \frac{1}{2} + \frac{1}{2} \times \frac{1}{2}$); but if they do the same with the three categories in

the second scheme, they would only agree on a third of the items ($\frac{1}{3} \times \frac{1}{3} + \frac{1}{3} \times \frac{1}{3} + \frac{1}{3} \times \frac{1}{3}$). Moreover, chance agreement varies with the distribution of items among categories: as a result, we may again observe a higher percentage agreement when one category is much more common than the other. This problem, already discussed by Hsu and Field (2003, page 207) among others, can be illustrated using the example in Figure 3 in Di Eugenio and Glass (2004, pages 98–99). Suppose 95% of utterances in a particular domain are acceptances, and only 5% are acknowledgments. We would then expect by chance that $.95 \times .95 = .9025$ of the utterances would be classified as acceptances by both coders, and $.05 \times .05 = .0025$ as acknowledgments, so the coders would agree on 90.5% of the utterances. Under such circumstances, a seemingly high observed agreement of 90% is actually worse than expected by chance.

In order to get figures that are comparable across studies, observed agreement has to be adjusted for chance agreement. We look at various ways of doing this starting in section 2.3. In the remainder of this section we look at other chance-adjusted measures and show why they are inadequate for measuring agreement, which is the motivation for developing specific agreement coefficients. We will not look at the variants of percentage agreement used in CL work on discourse before the introduction of kappa, such as percentage agreement with an expert and percentage agreement with the majority; see Carletta (1996) for discussion and criticism.

Measures of association The χ^2 statistic is also inappropriate as a measure of agreement. As pointed out by Cohen (1960, page 39), χ^2 is a measure of *association* – which means that we get a high value of χ^2 whenever a particular cooccurrence of judgments is different from the expected value. This may happen not just when we find good agreement, but also when we have systematic disagreement. The agreement matrix in Table 2 (adapted from Cohen 1960) reports the results of an annotation experiment in which again coder A and coder B classify utterances as either **Statements**, **Info-Requests**, or **Checks**. The value of χ^2 for this table is 64.59, which is highly significant; but the highest contribution comes from cell A-IREQ/B-CHCK, where the observed value .15 is much higher than the expected value .06. This cell specifies the percentage of utterances classified by A as Information Requests and by B as Checks: a case of disagreement.

		CODER A			
		STAT	IREQ	CHCK	TOTAL
CODER B	STAT	.25	.13	.12	.50
	IREQ	.12	.02	.16	.30
	CHCK	.03	.15	.02	.20
	TOTAL	.40	.30	.30	1

Table 2: High association but low agreement

Correlation Coefficients A point perhaps not sufficiently emphasized in the CL literature on agreement is that κ and related measures of agreement such as α or π are not primarily statistics in the sense of t, χ^2 or F, which are (functions associated with) probability distributions whose value specifies the significance of the result obtained. The title of Cohen's well-known article is very illuminating in this respect: π, κ, α, etc. are 'coefficient(s) of agreement for nominal scales'. What this means is that they are coefficients taking values between -1 and $+1$, just like Pearson's product-moment

coefficient r or Spearman's rank-correlation coefficient r_s, but intended for nominal scales, and for measuring agreement rather than association. Thinking of the kappa-like measures of agreement as coefficients is illuminating in certain respects, as they have some of the formal properties of correlation coefficients (Krippendorff 1970), and the problem of deciding whether a particular value of, say, κ indicates a sufficient degree of agreement is similar to the problem of determining whether a particular value of r expresses a strong enough association. However, neither product-moment correlation r not rank order correlation r_s are good measures of agreement (Bartko and Carpenter 1976, page 309). This is not just because these coefficients are specified for real values rather than nominal scales; the real problem is that correlation is not the same thing as agreement, even with numerical values, in that a strong correlation may exist even when coders disagree. The problem is illustrated by Table 3 (adapted from Bartko and Carpenter 1976). Suppose we have a coding scheme according to which coders give each item a rating between 1 and 10 (this might be a marking scheme for student essays, for example), and we ran two experiments to test the scheme. In the first experiment, coders A and B (whose marks are shown in columns 2 and 3) are in complete agreement; while in the second, coders C and D (whose marks are shown in columns 4 and 5) disagree on all items, but assign marks that are linearly correlated. Exactly the same product-moment value will be obtained in both experiments, even though there is perfect agreement between A and B, but no agreement at all between C and D.

Item	Exp 1		Exp 2	
	A	B	C	D
a	1	1	1	2
b	2	2	2	4
c	3	3	3	6
d	4	4	4	8
e	5	5	5	10
	$r = 1.0$		$r = 1.0$	

Table 3: Correlation need not indicate agreement

2.3 Chance-corrected coefficients for measuring agreement between two coders

All of the coefficients of agreement discussed in this paper correct for chance on the basis of the same idea. First we find how much agreement is expected by chance: let us call this value A_e. The value $1 - A_e$ will then measure how much agreement over and above chance is attainable; whereas the value $A_o - A_e$ will tell us how much agreement beyond chance was actually found. The ratio between $A_o - A_e$ and $1 - A_e$ will then tell us the proportion of the possible agreement beyond chance was actually observed. This idea is expressed by the following formula.

$$S, \pi, \kappa = \frac{A_o - A_e}{1 - A_e}$$

The three best-known coefficients of agreement, S (Bennett et al. 1954), π (Scott 1955) and κ (Cohen 1960), and their generalizations, all use this formula; whereas Krippendorff's α is based on an equivalent formula, although expressed in terms of disagreement (see section 2.5). All three coefficients therefore yield values of agreement between $-A_e/1 - A_e$ (no observed agreement) and 1 (observed agreement = 1).

The difference between S, π and κ is in the assumptions they make about expected agreement A_e, defined as the probability that two coders will classify an arbitrary item as belonging to the same category by chance. As explained perhaps most clearly by Krippendorff (1980), when trying to measure expected agreement we have to confront a problem that is all too familiar in CL. We do not have independent prior knowledge of the distribution of items among categories: A_e therefore has to be estimated from the observed data, making a variety of assumptions. All three coefficients are based on an *independence* assumption: that the two coders assigning tags by chance are acting independently – that is, that the chance of c_1 and c_2 agreeing on any given category k is the product of the chance of each of them assigning an item to that category: $P(k|c_1) \cdot P(k|c_2)$ (the independence assumption has been the subject of much criticism, for example by John S. Uebersax).[2] Expected agreement is then the probability of c_1 and c_2 agreeing on any category, i.e., the sum over all categories of the probabilities of the coders agreeing on a given category:

$$A_e^S = A_e^\pi = A_e^\kappa = \sum_{k \in K} P(k|c_1) \cdot P(k|c_2)$$

The difference between S, π and κ is the way they estimate $P(k|c_i)$ (Zwick 1988; Hsu and Field 2003). The simplest, yet strongest, assumption is used in the estimation of expected agreement for S: all categories are assumed to be uniformly distributed (i.e., equally likely). A weaker assumption is made when estimating π: expected agreement is estimated on the basis of the distribution of categories found in the observed data, but it is assumed that all coders assign categories on the basis of a single (but not necessarily uniform) distribution. In the estimation of κ, finally, no further assumption is made beyond independence: the expected agreement is directly computed from the observed distributions of the individual annotators. We begin here by considering the two-coder case, and discuss a variety of proposed generalizations starting in section 2.4.

All categories are equally likely: *S* The simplest way of discounting for chance is the one adopted to compute the coefficient S, also known in the literature as C, κ_n, G, and RE (see Zwick 1988; Hsu and Field 2003). As said above, the computation of S is based on a strong uniformity assumption – i.e., on the assumption that all categories are equally likely. If coders classify the items into $\mathbf{k}$ categories, then the chance $P(k|c_i)$ of any of them assigning an item to category k under the uniformity assumption is $\frac{1}{\mathbf{k}}$; hence the total agreement expected by chance is

$$A_e^S = \sum_{k \in K} P(k|c_1) \cdot P(k|c_2) = \mathbf{k} \cdot \left(\frac{1}{\mathbf{k}}\right)^2 = \frac{1}{\mathbf{k}}$$

For example, the value of S for the coding example in Table 1 (for which $A_o = .7$, see above) is as follows.

$$P(k|\text{Coder A}) = P(k|\text{Coder B}) = \frac{1}{2}$$

$$A_e^S = 2 \times \left(\frac{1}{2}\right)^2 = .5$$

$$S = \frac{.7 - .5}{1 - .5} = .4$$

[2] http://ourworld.compuserve.com/homepages/jsuebersax/agree.htm, accessed September 5, 2005.

Brennan and Prediger (1981) argue that this is the best assumption one can make about the way coders distribute items among categories if we have no independent prior knowledge of the distribution. However, the uniformity assumption is problematic in many respects. A problem already noted in Scott (1955, pages 322–323) is that the value of the coefficient can be artificially increased simply by adding spurious categories which the coders would never use. In the case of CL, for example, S would reward designing extremely fine-grained tagsets, provided that most tags are never actually encountered in real data. And anyway, the uniform distribution assumption isn't very plausible for CL, as in pretty much all tagging tasks, from POS tags (Francis and Kucera 1982) to wordsenses (Fellbaum et al. 1997; Bruce and Wiebe 1998) to dialogue acts (Carletta et al. 1997) we find significant differences in the distribution of tags. Additional limitations are noted by Hsu and Field (2003). For these reasons the S coefficient has never really found much use in CL, and studying it does not contribute to the points we develop in this paper, so we will not discuss it further.

A single distribution: π All of the methods for discounting chance agreement we discuss in this paper attempt to overcome the limitations of S's strong uniformity assumption using an idea first proposed by Scott (1955): use the actual behavior of the coders to estimate the prior distribution of the categories. As said above, Scott based his characterization of π on the assumption that a single probability distribution underlies the random assignment of categories to items for both coders, and proposed to estimate this distribution using $\hat{P}(k)$, the observed proportion of items assigned to category k by either coder.[3]

$$P(k|c_1) = P(k|c_2) = \hat{P}(k)$$

Given the assumption that coders act independently, A_e^π can then be computed as follows:

$$A_e^\pi = \sum_{k \in K} P(k|c_1) \cdot P(k|c_2) = \sum_{k \in K} \left(\hat{P}(k)\right)^2$$

$\hat{P}(k)$ is the total number of assignments to k by both coders $\mathbf{n}_k$, divided by the overall number of assignments, which for the two-coder case is twice the number of items $\mathbf{i}$:

$$\hat{P}(k) = \frac{1}{2\mathbf{i}} \mathbf{n}_k$$

We obtain in the end the following equation to compute A_e^π.

$$A_e^\pi = \sum_{k \in K} \left(\frac{1}{2\mathbf{i}} \mathbf{n}_k\right)^2$$

For example, the value of π for the experiment in Table 1 is as follows.

$$P(\text{Stat} \mid \text{Coder A}) = P(\text{Stat} \mid \text{Coder B}) = \hat{P}(\text{Stat}) = .35$$

$$P(\text{IReq} \mid \text{Coder A}) = P(\text{IReq} \mid \text{Coder B}) = \hat{P}(\text{IReq}) = .65$$

$$A_e^\pi = .35^2 + .65^2 = .1225 + .4225 = .545$$

$$\pi = \frac{.7 - .545}{1 - .545} = \frac{.155}{.455} \approx .341$$

[3]Notice that this is the method used to compute the K coefficient discussed by Siegel and Castellan (1988), which is why we consider K to be a generalization of π rather than κ, as already pointed out by Di Eugenio and Glass (2004).

Allowing for Bias: κ The method proposed by Cohen (1960) to estimate expected agreement A_e in his κ coefficient takes into account the way coders distribute items into categories, just like the method proposed by Scott for π. What differentiates κ from π is that in κ the so-called annotator **bias** is taken into account, as noted by Di Eugenio and Glass (2004) (and, in other fields, by Zwick 1988, Feinstein and Cicchetti 1990, and Cicchetti and Feinstein 1990). What this means is that the computation of $P(k|c)$ used in κ is not based on the overall distribution of items into categories, but on the individual coder c's proportion of items assigned to category k, $\hat{P}(k|c)$:

$$P(k|c) = \hat{P}(k|c)$$

$\hat{P}(k|c)$ is the number of assignments to k by c, $\mathbf{n}_{ck}$, divided by the number of items $\mathbf{i}$.

$$\hat{P}(k|c) = \frac{1}{\mathbf{i}} \mathbf{n}_{ck}$$

As in the case of S and π, the probability that the two coders c_1 and c_2 assign an item to a particular category $k \in K$ is the joint probability of each coder making this assignment independently. For κ this joint probability is $\hat{P}(k|c_1) \cdot \hat{P}(k|c_2)$; expected agreement is then the sum of this joint probability over all the categories $k \in K$.

$$A_e^\kappa = \sum_{k \in K} P(k|c_1) \cdot P(k|c_2) = \sum_{k \in K} \hat{P}(k|c_1) \cdot \hat{P}(k|c_2)$$

The value of κ for the experiment in Table 1 is as follows.

$$P(\text{Stat} \mid \text{Coder A}) = .3 \quad P(\text{Stat} \mid \text{Coder B}) = .4$$
$$P(\text{IReq} \mid \text{Coder A}) = .7 \quad P(\text{IReq} \mid \text{Coder B}) = .6$$
$$A_e^\kappa = .3 \times .4 + .6 \times .7 = .12 + .42 = .54$$
$$\kappa = \frac{.7 - .54}{1 - .54} = \frac{.16}{.46} \approx .348$$

The computation of A_e used for κ's is in at least two respects more intuitive than that used for Scott's π: first of all, it makes fewer assumptions; and secondly, it makes intuitive sense to treat differences in bias as disagreements. There are also a few problems, however, the most serious of which are the paradoxes observed by Cicchetti and Feinstein (Feinstein and Cicchetti 1990; Cicchetti and Feinstein 1990) and brought to the attention of the CL community by Di Eugenio and Glass (2004). We return to this issue after illustrating the differences between the coefficients with an example.

The effect of bias on the values of S, Pi and Kappa Zwick (1988) contains a particularly clear illustration of the effect of differences between the coders' probability distributions (so-called **coder marginals**) on the values of S, κ, and π. We reproduce one of her examples here, adapting her discussion to our purposes.

Let us assume a scheme, again adapted from DAMSL, with four categories for forward-looking function: **Statement**, **Info-Request**, **Influencing-Addressee-Future-Action**, and **Committing-Speaker-Future-Action**. Let us again assume we have two coders, coder A and coder B. In Table 4 we find three illustrations of the three situations that may arise, in all of which $A_o = .60$.

Case 1 is an example of the case in which the coders assign equal proportions of items to all categories; in this case, all three coefficients of agreement have the same value. Case 2 is an example of the situation in which coder A and coder B, while not assigning equal proportions of items to all

		Coder A				
		Stat	IReq	Iafa	Csfa	Total
	Case 1: Marginals uniform $S = .467, \pi = .467, \kappa = .467$					
Coder B	Stat	.20	–	–	.05	.25
	IReq	–	.10	.15	–	.25
	Iafa	–	.15	.10	–	.25
	Csfa	.05	–	–	.20	.25
	Total	.25	.25	.25	.25	1.00
	Case 2: Marginals equal but not uniform $S = .467, \pi = .444, \kappa = .444$					
Coder B	Stat	.20	.10	.10	–	.40
	IReq	.10	.10	–	–	.20
	Iafa	.10	–	.10	–	.20
	Csfa	–	–	–	.20	.20
	Total	.40	.20	.20	.20	1.00
	Case 3: Marginals unequal $S = .467, \pi = .460, \kappa = .474$					
Coder B	Stat	.20	.05	.05	.10	.40
	IReq	–	.10	.05	.05	.20
	Iafa	–	.05	.10	.05	.20
	Csfa	–	–	–	.20	.20
	Total	.20	.20	.20	.40	1.00

Table 4: The effect of coder marginals on coefficient values

categories, still end up assigning items to categories in equal proportions: both judge 40% of items to be **Statements**, 20% to be **Info-Requests**, and so forth. In this situation, κ and π still have the same value. Finally, Case 3 is an example of the situation in which Coder A and Coder B do not even agree on the proportion of items belonging to a given category: in this case, κ and π may have different values. Notice also that in Case 2, we get lower values of κ and π than in Case 3 – that is, the coders get penalized for agreeing on the marginals (Feinstein and Cicchetti 1990; Cicchetti and Feinstein 1990; Di Eugenio and Glass 2004).

2.4 More than two coders

The definitions of κ and π presented above are for the case of two coders, but in corpus annotation practice, measuring reliability with only two coders is seldom considered enough, except for small-scale studies. A coefficient of agreement for multiple coders was proposed by Fleiss (1971), who called it κ even though it assumes a single probability distribution for all coders and is thus better thought of as a generalization of Scott's π. This unfortunate choice of name was the cause of much confusion in subsequent literature: often, discussions of generalizations of κ to more than two coders actually report Fleiss's coefficient (e.g. Bartko and Carpenter 1976; Siegel and Castellan 1988; Di Eugenio and Glass 2004). We will call this coefficient multi-π, dropping the multi- prefix when no confusion is expected to arise. We will reserve the name multi-κ for a generalization of Cohen's κ proper which we will introduce below, due to Davies and Fleiss (1982).

Fleiss's multi-π When the number of coders **c** is greater than two it is no longer possible to represent coder judgments using a simple contingency table like Table 1 or Table 2, since each coder has to be represented in a separate dimension. Fleiss (1971) therefore uses a different type of table which lists each item with the number of judgments it received for each category; Siegel and Castellan (1988) use a similar table, which Di Eugenio and Glass (2004) call an **agreement table**. Table 5 is an example of such an agreement table, in which the same 100 utterances from Table 1 are labeled using three coders instead of two.

	STAT	IREQ
Utt_1	2	1
Utt_2	0	3
⋮		
Utt_{100}	1	2
TOTAL	90 (.3)	210 (.7)

Table 5: Agreement table with three coders.

Di Eugenio and Glass (2004, page 97) note that compared to contingency tables like Tables 1 and 2, agreement tables like Table 5 lose information because they do not say which coder gave each judgment. This information is not used in the calculation of π, but is necessary for determining the individual coders' distributions in the calculation of κ. (Agreement tables also add information compared to contingency tables, namely the identity of the items that make up each contingency class, but this information is not used in the calculation of either κ or π.)

With more than two coders, the observed agreement A_o can no longer be defined as the percentage of items on which there is agreement, since inevitably there will be items on which some coders agree and others disagree. The amount of agreement on a particular item is therefore defined by Fleiss (1971) as the proportion of agreeing judgment pairs out of the total number of judgment pairs for that item.

Let $\mathbf{n}_{ik}$ stand for the number of times an item i is classified in category k (i.e. the number of coders that make such a judgment): for example, given the distribution in Table 5, $\mathbf{n}_{1\text{Stat}} = 2$ and $\mathbf{n}_{1\text{IReq}} = 1$. Each category k contributes $\binom{\mathbf{n}_{ik}}{2}$ pairs of agreeing judgments for item i; the amount of agreement agr_i for item i is the sum of $\binom{\mathbf{n}_{ik}}{2}$ over all categories $k \in K$, divided by $\binom{\mathbf{c}}{2}$, the total number of judgment pairs per item.

$$\text{agr}_i = \frac{1}{\binom{\mathbf{c}}{2}} \sum_{k \in K} \binom{\mathbf{n}_{ik}}{2} = \frac{1}{\mathbf{c}(\mathbf{c}-1)} \sum_{k \in K} \mathbf{n}_{ik}(\mathbf{n}_{ik}-1)$$

For example, given the results in Table 5,

$$\text{agr}_1 = \frac{1}{\binom{3}{2}} \left(\binom{\mathbf{n}_{1\text{Stat}}}{2} + \binom{\mathbf{n}_{1\text{IReq}}}{2} \right) = \frac{1}{3 \times 2} (2 + 0) = \frac{2}{6} \approx 0.33$$

The overall observed agreement is the mean of agr_i for all items $i \in I$.

$$A_o = \frac{1}{\mathbf{i}} \sum_{i \in I} \text{agr}_i = \frac{1}{\mathbf{ic}(\mathbf{c}-1)} \sum_{i \in I} \sum_{k \in K} \mathbf{n}_{ik}(\mathbf{n}_{ik}-1)$$

(Notice that the above definition of observed agreement is equivalent to the mean of the two-coder obeserved agreement values from section 2.3 for all coder pairs.)

If observed agreement is measured on the basis of pairwise agreement (the proportion of agreeing judgment pairs), it makes sense to measure *expected* agreement in terms of pairwise comparisons as well, i.e., as the probability that any pair of judgments for an item would be in agreement – or, said otherwise, the probability that two arbitrary coders would make the same judgment for a particular item by chance. This is the approach taken by Fleiss (1971) to define his 'κ', multi-π. Like Scott, Fleiss assumes that the behavior of coders can be described with a single probability distribution, meaning that expected agreement is calculated using $\hat{P}(k)$, the overall probability of assigning an item to category k, which just as in the case of Scott's π, is the total number of such assignments by all coders $\mathbf{n}_k$ divided by the overall number of assignments. The latter, in turn, is the number of items $\mathbf{i}$ multiplied by the number of coders $\mathbf{c}$.

$$\hat{P}(k) = \frac{1}{\mathbf{ic}} \mathbf{n}_k$$

The probability that two arbitrary coders assign an item to a particular category $k \in K$ is assumed to be the joint probability of each coder making this assignment independently. For π, this joint proability is $\hat{P}(k)^2$. The expected agreement is the sum of this joint probability over all the categories $k \in K$.

$$A_e^{\pi} = \sum_{k \in K} \hat{P}(k)^2$$

This is the coefficient that Siegel and Castellan (1988) call K.

Multi-κ It is fairly straightforward to adapt Fleiss's proposal to generalize Cohen's κ proper for more than two coders; the development below is our own, but an identical proposal can be found in Davies and Fleiss (1982).

For multi-κ, we calculate a separate probability distribution for each annotator: $\hat{P}(k|c)$ is the probability of assigning an item to category k by coder c, which is the number of such assignments $\mathbf{n}_{ck}$ divided by the number of items $\mathbf{i}$.

$$\hat{P}(k|c) = \frac{1}{\mathbf{i}}\,\mathbf{n}_{ck}$$

Again, the probability that two arbitrary coders assign an item to a particular category $k \in K$ is the joint probability of each coder making this assignment independently. For multi-κ, the joint probability for two particular coders c_m and c_n is $\hat{P}(k|c_m)\hat{P}(k|c_n)$, and since all coders judge all items, the joint probability for an arbitrary pair of coders is the arithmetic mean of $\hat{P}(k|c_m)\hat{P}(k|c_n)$ over all coder pairs c_m, c_n. Again, the expected agreement is the sum of this joint probability over all the categories $k \in K$.

$$A_e^\kappa = \sum_{k\in K} \frac{1}{\binom{\mathbf{c}}{2}} \sum_{m=1}^{\mathbf{c}-1} \sum_{n=m+1}^{\mathbf{c}} \hat{P}(k|c_m)\hat{P}(k|c_n)$$

It is easy to see that A_e^κ for multiple coders is the mean of the two-coder A_e^κ values from section 2.3 for all coder pairs.

2.5 Weighted agreement coefficients

A serious limitation of both π and κ is that they treat all disagreements equally. For some coding tasks, however, disagreements are not all alike. Even in the simpler case of dialogue act tagging, a disagreement between an **Accept** and a **Reject** interpretation of an utterance is clearly more serious than a disagreement between an **Info-Request** and a **Check**; for tasks such as anaphora resolution, in which the most plausible form of tagging proposed so far uses sets (coreference chains) as labels, allowing for degrees of disagreement becomes essential (see section 4 for a discussion of NLP tasks that fall under this characterization). Under such circumstances π and κ yield extremely low values and are thus not very useful; instead, what is needed are coefficients that can take into account the magnitude of the disagreements. In this section we look at two such coefficients – α (Krippendorff 1980) and weighted kappa κ_w (Cohen 1968). The relation between the two is similar to that between π and κ, in that the former assumes that expected agreement is the result of a single probability distribution while the latter assumes a separate distribution for each coder. Additionally, α applies to multiple coders while κ_w is restricted to two coders (for a generalization of κ_w to multiple coders see section 2.7 below).

Both α and κ_w differ from the coefficients seen so far in that their computation involves determining *disagreements* rather than agreements. This is because when disagreements are of different magnitudes it is not sufficient to count the agreeing judgment pairs – the pairs on the diagonal – as we did for A_o; we should also count and weigh the disagreeing pairs. It is thus more convenient (though by no means necessary) to measure observed and expected disagreement rather than agreement; the result is then subtracted from 1 to yield a final value of agreement.

$$\kappa_w, \alpha = 1 - \frac{D_o}{D_e}$$

As the following equivalence shows, this formula is equivalent to that of π and κ.

$$1 - \frac{D_o}{D_e} = 1 - \frac{1 - A_o}{1 - A_e} = \frac{1 - A_e - (1 - A_o)}{1 - A_e} = \frac{A_o - A_e}{1 - A_e}$$

The crucial difference between the coefficients discussed in this section and those discussed before is that the computation of weighted coefficients involves a **distance metric**: a function **d** from category pairs to non-negative real numbers that specifies the amount of dissimilarity between the categories. The appropriate metric for an individual coding task is determined by the nature of the categories. Cohen (1968) does not place any general constraints on the distance metric (**v** in his notation), but we will adopt two constraints that seem appropriate as general characterizations of distance (all of the distance metrics in Krippendorff 1980 conform to these constraints).

1. For every category $k \in K$, $\mathbf{d}_{kk} = 0$ (the distance between a category and itself is minimal).
2. For every two categories $k_a, k_b \in K$, $\mathbf{d}_{k_a k_b} = \mathbf{d}_{k_b k_a}$ (the distance between two categories does not depend on their order).

The computation of observed disagreement for α and κ_w, much like the computation of observed agreement for the previously introduced coefficients, is an average of the disagreement values over the specific items, except that, as said above, *all* disagreement values are considered, not only those on the diagonal. Again as in the case of observed agreement, it involves measuring pairs of judgments: the amount of disagreement for each item $i \in I$ is the arithmetic mean of the distances between the pairs of judgments pertaining to it. Let $\mathbf{n}_{ik}$ stand for the number of times item i is judged to belong to category k (i.e. the number of coders that make such judgment). For every pair of distinct categories $k_a, k_b \in K$ there are $\mathbf{n}_{ik_a}\mathbf{n}_{ik_b}$ pairs of judgments of item i, each contributing a distance of $\mathbf{d}_{k_a k_b}$. The mean disagreement for item i is the sum of these distances over all category pairs, divided by $\mathbf{c}(\mathbf{c}-1)$, the total number of ordered judgment pairs for the item. (Note that the formula below incorrectly counts the number of pairs of identical judgments; there's no need to correct for this because identical pairs contribute a distance of zero.)

$$\text{disagr}_i = \frac{1}{\mathbf{c}(\mathbf{c}-1)} \sum_{j=1}^{\mathbf{k}} \sum_{l=1}^{\mathbf{k}} \mathbf{n}_{ik_j}\mathbf{n}_{ik_l}\mathbf{d}_{k_j k_l}$$

The overall observed disagreement is the arithmetic mean of the values disagr_i for all items $i \in I$.

$$\text{D}_\text{o} = \frac{1}{\mathbf{ic}(\mathbf{c}-1)} \sum_{i \in I} \sum_{j=1}^{\mathbf{k}} \sum_{l=1}^{\mathbf{k}} \mathbf{n}_{ik_j}\mathbf{n}_{ik_l}\mathbf{d}_{k_j k_l}$$

(Notice that observed disagreement is not measured in percentages, but rather in the units of the distance function. If we take all disagreements to be of equal weight, that is $\mathbf{d}_{k_a k_b} = 1$ for all $k_a \neq k_b$, then the observed disagreement is exactly the complement of the observed agreement as calculated in section 2.4: $\text{D}_\text{o} = 1 - \text{A}_\text{o}$.)

The disagreement expected by chance is also a measure of distance: the expected distance for an arbitrary judgment pair, that is, for a random assignment of categories by two coders. This, in turn, is the arithmetic mean of all possible distances between category pairs, with each distance weighted by the probability that two arbitrary coders would choose that particular category pair. But because of the different assumptions about this probability distribution made in α and κ_w, the computation of D_e differs for the two coefficients, just as it differed between π and κ.

(A slight variant of) Krippendorff's α We start with a slight variation of Krippendorff's α which we will call α'. (The reasons for introducing a variant will hopefully become clear shortly.) With α, as with π, the expected disagreement is assumed to be the result of a single probability distribution

characterizing the beliefs of all coders. The overall probability of assigning an item to category k, $\hat{\mathrm{P}}(k)$, is the total number of such assignments by all coders $\mathbf{n}_k$ divided by the overall number of assignments, which is the number of items $\mathbf{i}$ multiplied by the number of coders $\mathbf{c}$.

$$\hat{\mathrm{P}}(k) = \frac{1}{\mathbf{ic}}\,\mathbf{n}_k$$

Given an arbitrary pair of coders, the probability that the first assigns an item to category k_a and the second to category k_b is the joint probability of each coder making this assignment independently, namely $\hat{\mathrm{P}}(k_a)\hat{\mathrm{P}}(k_b)$. The expected disagreement is the mean of the distances between categories, weighted by these probabilities for all (ordered) category pairs.

$$\mathrm{D}_{\mathrm{e}}^{\alpha'} = \sum_{j=1}^{\mathbf{k}}\sum_{l=1}^{\mathbf{k}} \hat{\mathrm{P}}(k_j)\hat{\mathrm{P}}(k_l)\mathbf{d}_{k_jk_l} = \frac{1}{(\mathbf{ic})^2}\sum_{j=1}^{\mathbf{k}}\sum_{l=1}^{\mathbf{k}} \mathbf{n}_{k_j}\mathbf{n}_{k_l}\mathbf{d}_{k_jk_l}$$

If we take all disagreements to be of equal weight, that is $\mathbf{d}_{k_ak_b} = 1$ for all $k_a \neq k_b$, then this measure of expected disagreement is exactly the complement of the expected agreement for π as calculated in section 2.4: $\mathrm{D}_{\mathrm{e}}^{\alpha'} = 1 - \mathrm{A}_{\mathrm{e}}^{\pi}$.

The difference between Krippendorff's α and our α' is that Krippendorff's definition of the expected disagreement is slightly different from ours – it is the mean of the distances between all the judgment pairs in the data, without regard to items. This comes out as the following.

$$\mathrm{D}_{\mathrm{e}}^{\alpha} = \frac{1}{\mathbf{ic}(\mathbf{ic}-1)}\sum_{j=1}^{\mathbf{k}}\sum_{l=1}^{\mathbf{k}} \mathbf{n}_{k_j}\mathbf{n}_{k_l}\mathbf{d}_{k_jk_l}$$

On the relation between α and π, Krippendorff notes that "when the number of coders is exactly two, the categories of the variables are unordered (nominal scale), and the sample size is very large, then our agreement coefficient equals Scott's (1955) pi ... Our coefficient corrects for small sample sizes" (page 138). Indeed, we saw that our coefficient α' is equivalent to π when when all disagreements are considered equal, and it is easy to see that α and α' approach each other when either the number of items or the number of coders is very large.

$$\lim_{\mathbf{i}\to\infty \text{ or } \mathbf{c}\to\infty}\left(\frac{1}{\mathbf{ic}(\mathbf{ic}-1)}\sum_{j=1}^{\mathbf{k}}\sum_{l=1}^{\mathbf{k}} \mathbf{n}_{k_j}\mathbf{n}_{k_l}\mathbf{d}_{k_jk_l}\right) = \frac{1}{(\mathbf{ic})^2}\sum_{j=1}^{\mathbf{k}}\sum_{l=1}^{\mathbf{k}} \mathbf{n}_{k_j}\mathbf{n}_{k_l}\mathbf{d}_{k_jk_l}$$

It is not clear to us what assumptions underlie Krippendorff's calculation of the expected disagreement, and why he considers his calculation to be more correct.

Cohen's κ_w The coefficient κ_w assumes, as for κ, that the expected disagreement is the result of a distinct probability distribution for each coder: $\hat{\mathrm{P}}(k|c)$ is the probability of assigning an item to category k by coder c, that is the number of such assignments $\mathbf{n}_{ck}$ divided by the number of items $\mathbf{i}$.

$$\hat{\mathrm{P}}(k|c) = \frac{1}{\mathbf{i}}\,\mathbf{n}_{ck}$$

Weighted kappa κ_w is restricted to two coders; the probability that the first assigns an item to category k_a and the second to category k_b is the joint probability of each coder making this assignment independently, namely $\hat{\mathrm{P}}(k_a|c_1)\hat{\mathrm{P}}(k_b|c_2)$. The expected disagreement is the mean of the distances between categories, weighted by these probabilities for all (ordered) category pairs. (In Cohen 1968,

both observed and expected disagreement are normalized to the interval $[0, 1]$ through division by the maximal distance; this is redundant, since in the final equation the observed disagreement is divided by the expected disagreement.)

$$\mathrm{D}_\mathrm{e}^{\kappa_w} = \sum_{j=1}^{\mathbf{k}} \sum_{l=1}^{\mathbf{k}} \hat{\mathrm{P}}(k_j|c_1)\hat{\mathrm{P}}(k_l|c_2)\mathbf{d}_{k_j k_l} = \frac{1}{\mathbf{i}^2} \sum_{j=1}^{\mathbf{k}} \sum_{l=1}^{\mathbf{k}} \mathbf{n}_{c_1 k_j}\mathbf{n}_{c_2 k_l}\mathbf{d}_{k_j k_l}$$

If we take all disagreements to be of equal weight, that is $\mathbf{d}_{k_a k_b} = 1$ for all $k_a \neq k_b$, then this measure of expected disagreement is exactly the complement of the expected agreement for κ as calculated in section 2.3: $\mathrm{D}_\mathrm{e}^{\kappa_w} = 1 - \mathrm{A}_\mathrm{e}^{\kappa}$.

2.6 The Coefficient Cube

The agreement coefficients we have seen can all be thought of as generalizations of Scott's π along three different dimensions. One dimension is the calculation of expected agreement using separate probability distributions for the individual coders, as done by κ. Another dimension is a generalization from two coders to multiple coders, resulting in multi-π (Fleiss's κ) and multi-κ (Davies and Fleiss 1982). A third dimension is the introduction of weighted agreement coefficients – α' for multiple coders with a single distribution, and κ_w for two coders with separate distributions. The relations between the various coefficients are depicted in Figure 1.

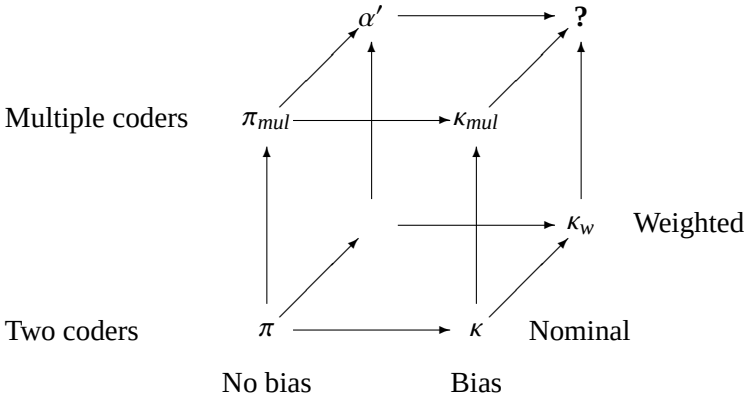

Figure 1: Generalizing π along three dimensions

What is missing from the picture is a coefficient that generalizes π along all three dimensions – an agreement coefficient that is weighted, applies to multiple coders, and calculates expected agreement using a separate probability distribution for each coder.

2.7 Filling the gap: β

Such a coefficient can be thought of as a generalization of κ_w to multiple coders, or alternatively as a generalization of α' which uses individual coders' distributions. We now develop such a coefficient, calling it β which should serve as a reminder that it is like α with bias.

Like the other weighted coeficients, β measures the observed and expected disagreement, whose ratio is subtracted from one.

$$\beta = 1 - \frac{\mathrm{D_o}}{\mathrm{D_e}}$$

The observed disagreement is the same as for the other weighted measures, that is the mean disagreement per item, where the disagreement per item is the mean distance between all the judgment pairs pertaining to it (section 2.5).

The expected disagreement is the expected distance for an arbitrary judgment pair, which is the arithmetic mean of all possible distances between category pairs weighted by the probabilities for choosing particular pairs. We estimate the probability that coder c assigns an item to category k as the total number of such assignments $\mathbf{n}_{ck}$ divided by the overall number of assignments for this coder, which is the number of items $\mathbf{i}$.

$$\hat{\mathrm{P}}(k|c) = \frac{1}{\mathbf{i}}\,\mathbf{n}_{ck}$$

The probability that two particular coders c_m and c_n assign an item to two distinct categories k_a and k_b is $\hat{\mathrm{P}}(k_a|c_m)\hat{\mathrm{P}}(k_b|c_n) + \hat{\mathrm{P}}(k_b|c_m)\hat{\mathrm{P}}(k_a|c_n)$. Since all coders judge all items, the probability that an arbitrary pair of coders assign an item to k_a and k_b is the arithmetic mean of $\hat{\mathrm{P}}(k_a|c_m)\hat{\mathrm{P}}(k_b|c_n) + \hat{\mathrm{P}}(k_b|c_m)\hat{\mathrm{P}}(k_a|c_n)$ over all coder pairs.

$$\begin{aligned}
\hat{\mathrm{P}}(k_a, k_b) &= \frac{1}{\binom{\mathbf{c}}{2}} \sum_{m=1}^{\mathbf{c}-1} \sum_{n=m+1}^{\mathbf{c}} \hat{\mathrm{P}}(k_a|c_m)\hat{\mathrm{P}}(k_b|c_n) + \hat{\mathrm{P}}(k_b|c_m)\hat{\mathrm{P}}(k_a|c_n) \\
&= \frac{1}{\mathbf{i}^2\binom{\mathbf{c}}{2}} \sum_{m=1}^{\mathbf{c}-1} \sum_{n=m+1}^{\mathbf{c}} \mathbf{n}_{c_m k_a}\mathbf{n}_{c_n k_b} + \mathbf{n}_{c_m k_b}\mathbf{n}_{c_n k_a}
\end{aligned}$$

The expected disagreement is the mean of the distances for all distinct category pairs, weighted by the above probabilities (recall that identical category pairs contribute a distance of zero, so it does not matter if and how they are counted).

$$\begin{aligned}
\mathrm{D}_{\mathrm{e}}^{\beta} &= \sum_{j=1}^{\mathbf{k}-1} \sum_{l=j+1}^{\mathbf{k}} \hat{\mathrm{P}}(k_j, k_l)\mathbf{d}_{k_j k_l} \\
&= \frac{1}{\binom{\mathbf{c}}{2}} \sum_{j=1}^{\mathbf{k}-1} \sum_{l=j+1}^{\mathbf{k}} \sum_{m=1}^{\mathbf{c}-1} \sum_{n=m+1}^{\mathbf{c}} \Big(\hat{\mathrm{P}}(k_j|c_m)\hat{\mathrm{P}}(k_l|c_n) + \hat{\mathrm{P}}(k_l|c_m)\hat{\mathrm{P}}(k_j|c_n)\Big)\mathbf{d}_{k_j k_l} \\
&= \frac{1}{\binom{\mathbf{c}}{2}} \sum_{j=1}^{\mathbf{k}} \sum_{l=1}^{\mathbf{k}} \sum_{m=1}^{\mathbf{c}-1} \sum_{n=m+1}^{\mathbf{c}} \hat{\mathrm{P}}(k_j|c_m)\hat{\mathrm{P}}(k_l|c_n)\mathbf{d}_{k_j k_l} \\
&= \frac{1}{\mathbf{i}^2\binom{\mathbf{c}}{2}} \sum_{j=1}^{\mathbf{k}} \sum_{l=1}^{\mathbf{k}} \sum_{m=1}^{\mathbf{c}-1} \sum_{n=m+1}^{\mathbf{c}} \mathbf{n}_{c_m k_j}\mathbf{n}_{c_n k_l}\mathbf{d}_{k_j k_l}
\end{aligned}$$

It is easy to see that $\mathrm{D}_{\mathrm{e}}^{\beta}$ is the mean of the $\mathrm{D}_{\mathrm{e}}^{\kappa_w}$ values (section 2.5) over all coder pairs. If we take all disagreements to be of equal weight, that is $\mathbf{d}_{k_a k_b} = 1$ for all $k_a \neq k_b$, then this measure of expected disagreement is exactly the complement of the expected agreement for κ as calculated in section 2.4: $\mathrm{D}_{\mathrm{e}}^{\beta} = 1 - \mathrm{A}_{\mathrm{e}}^{\kappa}$.

2.8 An Integrated Example

We end this section with an example illustrating how all of the agreement coefficients discussed above are computed. To facilitate comparisons, all computations will be based on the annotation statistics in Table 6. This confusion matrix reports the results of an experiment where two coders classify a set of utterances into three categories.

		Coder A			
		Stat	IReq	Chck	Total
Coder B	Stat	46	6	0	52
	IReq	0	32	0	32
	Chck	0	6	10	16
	Total	46	44	10	100

Table 6: An integrated coding example

The unweighted coefficients Observed agreement for all of the unweighted coefficients–S, κ, and π– is calculated by counting the items on which the coders agree (the figures on the diagonal of the confusion matrix in Table 6) and dividing by the total number of items.

$$A_o = \frac{46 + 32 + 10}{100} = 0.88$$

Expected agreement for S is the reciprocal of the number of categories, or $\frac{1}{3}$; S is the observed agreement, discounted by this fraction.

$$\begin{aligned} A_e^S &= \frac{1}{3} \\ S &= \frac{A_o - A_e^S}{1 - A_e^S} = \frac{.88 - \frac{1}{3}}{1 - \frac{1}{3}} = .82 \end{aligned}$$

Expected agreement for π is the sum over all categories of the square of the mean of the individual coders' proportions; π is the observed agreement, discounted by this value.

$$\begin{aligned} A_e^\pi &= \left(\frac{46 + 52}{2 \times 100}\right)^2 + \left(\frac{44 + 32}{2 \times 100}\right)^2 + \left(\frac{10 + 16}{2 \times 100}\right)^2 = .49^2 + .38^2 + .13^2 = .4014 \\ \pi &= \frac{A_o - A_e^\pi}{1 - A_e^\pi} = \frac{.88 - .4014}{1 - .4014} \approx .7995 \end{aligned}$$

Expected agreement for κ is the sum over all categories of the product of the individual coders' proportions; κ is the observed agreement, discounted by this value.

$$\begin{aligned} A_e^\kappa &= \frac{46}{100} \times \frac{52}{100} + \frac{44}{100} \times \frac{32}{100} + \frac{10}{100} \times \frac{16}{100} = .396 \\ \kappa &= \frac{A_o - A_e^\kappa}{1 - A_e^\kappa} = \frac{.88 - .396}{1 - .396} \approx .8013 \end{aligned}$$

We see that the values of π and κ are very similar, which is to be expected when agreement is high, since this implies similar marginals. Notice however that $A_e^\kappa < A_e^\pi$, hence $\kappa > \pi$; this reflects a general property of κ and π, which will be discussed in section 3.

Weighted coefficients Suppose we notice that while **Statements** and **Info-Requests** are clearly distinct classifications, **Checks** are somewhere between the two. We therefore opt to weigh the distances

between the categories as follows (recall that 1 denotes maximal disagreement, and identical categories are in full agreement and thus have a distance of 0).

Statement–Statement:	0	Statement–Info-Request:	1
Info-Request–Info-Request:	0	Statement–Check:	.5
Check–Check:	0	Info-Request–Check:	.5

The observed disagreement is calculated by summing up *all* the cells in the contingency table, multiplying each cell by its respective weight, and dividing the total by the number of items (in the calculation below we ignore cells with zero items).

$$D_o = \frac{46\times 0 + 6\times 1 + 32\times 0 + 6\times .5 + 10\times 0}{100} = \frac{6+3}{100} = .09$$

The only sources of disagreement in the coding example of Table 6 are the six utterances marked as **Info-Requests** by coder A and **Statements** by coder B, which receive their full weight, and the six utterances marked as **Info-Requests** by coder A and **Checks** by coder B, which are given half their weight.

Expected disagreement for α' is the sum over all category pairs of the product of the mean of the individual coders' proportions, weighted by the distance; α' is the observed disagreement, discounted by this value.

$$\begin{aligned} D_e^{\alpha'} &= \tfrac{46+52}{2\times100}\times\tfrac{46+52}{2\times100}\times 0 + \tfrac{44+32}{2\times100}\times\tfrac{46+52}{2\times100}\times 1 + \tfrac{10+16}{2\times100}\times\tfrac{46+52}{2\times100}\times .5 \\ &\quad + \tfrac{46+52}{2\times100}\times\tfrac{44+32}{2\times100}\times 1 + \tfrac{44+32}{2\times100}\times\tfrac{44+32}{2\times100}\times 0 + \tfrac{10+16}{2\times100}\times\tfrac{44+32}{2\times100}\times .5 \\ &\quad + \tfrac{46+52}{2\times100}\times\tfrac{10+16}{2\times100}\times .5 + \tfrac{44+32}{2\times100}\times\tfrac{10+16}{2\times100}\times .5 + \tfrac{10+16}{2\times100}\times\tfrac{10+16}{2\times100}\times 0 \\ &= 2\times .49\times .38 + 2\times .49\times .13\times .5 + 2\times .38\times .13\times .5 = .4855 \\ \alpha' &= 1 - \frac{D_o}{D_e^{\alpha'}} = 1 - \frac{.09}{.4855} \approx .8146 \end{aligned}$$

Expected disagreement for Krippendorff's original α is similar to that of α', except that the denominator is $\mathbf{ik}(\mathbf{ik}-1)$, or $2\times100\times(2\times100-1) = 39800$; α is the observed disagreement, discounted by this value.

$$\begin{aligned} D_e^{\alpha} &= \tfrac{(46+52)\times(46+52)}{39800}\times 0 + \tfrac{(44+32)\times(46+52)}{39800}\times 1 + \tfrac{(10+16)\times(46+52)}{39800}\times .5 \\ &\quad + \tfrac{(46+52)\times(44+32)}{39800}\times 1 + \tfrac{(44+32)\times(44+32)}{39800}\times 0 + \tfrac{(10+16)\times(44+32)}{39800}\times .5 \\ &\quad + \tfrac{(46+52)\times(10+16)}{39800}\times .5 + \tfrac{(44+32)\times(10+16)}{39800}\times .5 + \tfrac{(10+16)\times(10+16)}{39800}\times 0 \\ &= \frac{1}{39800}\times(2\times98\times76 + 2\times98\times26\times.5 + 2\times76\times26\times.5) \approx .4879 \\ \alpha &= 1 - \frac{D_o}{D_e^{\alpha}} \approx 1 - \frac{.09}{.4879} \approx .8156 \end{aligned}$$

Finally, expected disagreement for β is the sum over all category pairs of the products of the individual coders' proportions, weighted by the distance; β is the observed disagreement, discounted by this value.

$$\begin{aligned} D_e^{\beta} &= \tfrac{46}{100}\times\tfrac{52}{100}\times 0 + \tfrac{44}{100}\times\tfrac{52}{100}\times 1 + \tfrac{10}{100}\times\tfrac{52}{100}\times .5 \\ &\quad + \tfrac{46}{100}\times\tfrac{32}{100}\times 1 + \tfrac{44}{100}\times\tfrac{32}{100}\times 0 + \tfrac{10}{100}\times\tfrac{32}{100}\times .5 \\ &\quad + \tfrac{46}{100}\times\tfrac{16}{100}\times .5 + \tfrac{44}{100}\times\tfrac{16}{100}\times .5 + \tfrac{10}{100}\times\tfrac{16}{100}\times 0 \end{aligned}$$

$$= .49$$

$$\beta = 1 - \frac{D_o}{D_e^{\beta}} = 1 - \frac{.09}{.49} \approx .8163$$

3 Bias, Revisited

The difference between the coefficients on the 'left' face and the 'right' face of the coefficient cube lies in the assumptions about what determines the agreement expected by chance, whether it is a single probability distribution (π) or separate probability distributions (κ); to these we may add S, which assumes a uniform distribution. Di Eugenio and Glass (2004) point out that the different assumptions reflect distinct conceptualizations of the reliability problem: the single distribution assumption considers differences between the coders' actual distributions of judgments to be noise in the data, whereas the individual distributions assumption considers such differences to reflect the relative biases of the individual coders, which is a genuine source of disagreement (Cohen 1960, pages 40–41). The question of the appropriateness of the different coefficients has been the subject of much debate in the literature; we will review the arguments and add some of our own.

Zwick (1988) suggests that the individual coders' distributions should be tested to see if the discrepancies are likely to be the result of random error or systematic disagreement; she proposes using the modified χ^2 test of Stuart (1955) for this purpose. If significant systematic disagreements are observed, then the data set should be considered unreliable; otherwise, the difference between π and κ will be small anyway, and Zwick recommends using π because it smooths over any remaining discrepancies.

In a criticism of Zwick's recommendation, Hsu and Field (2003) demonstrate how κ can give useful information even when the individual coders' distributions are very different. Likewise, Di Eugenio and Glass (2004, page 96) recommend using κ in "the vast majority ... of discourse- and dialogue-tagging efforts", where the individual coders' distributions tend to vary. Yet, Di Eugenio and Glass conclude that π should be reported too, because κ suffers from what they call the bias problem, described as "the paradox that κ_{Co} [our κ] increases as the coders become less similar" (page 99). Similar reservations about the use of κ have been noted by Brennan and Prediger (1981) and Zwick (1988).

We feel that the bias problem is less paradoxical than it sounds. While it is true that for a fixed observed agreement, a higher bias implies a lower expected agreement and therefore a higher κ value, the conclusion that κ penalizes coders for having similar distributions is unwarranted. This is because the observed agreement and expected agreement are not independent: both are drawn from the same set of observations. And if it so happens that two data sets have similar observed agreement and different biases, as in Table 4 above, then the data set with the higher bias is indeed more reliable. A high bias may indicate that the coding process is defective, but it also indicates that whatever data are agreed upon are less likely to be the result of chance errors. It is the latter point – the reliability of the data – that is captured by κ and other biased measures. This is the same logic by which a conclusion agreed upon by people with differing predispositions is more convincing than one agreed upon by people with similar predispositions, all else being equal.

A different objection to biased measures comes from Craggs and McGee Wood (in press), who argue that unbiased measures like π and α are preferable on methodological grounds. According to Craggs and McGee Wood, the purpose of an agreement coefficient is not to assess the reliability of the data, but rather that of the coding process. As such, the measure should be independent of the actual coders and should abstract over individual biases. In order to overcome individual biases, Craggs and

McGee Wood suggest increasing the number of coders, though without formal proof of the effects of such a move. As we will see shortly below, increasing the number of coders makes biased coefficients similar to unbiased ones.

We should point out that in practice the difference between π and κ doesn't often amount to much (see discussion in section 4). Moreover, one would expect the difference to diminish as the number of coders grows. We can make these points more precise, particularly the last one, by introducing a way of quantifying the difference between π and κ. This can be done as follows.

We define B, the overall **bias** in a particular set of coding data, as the difference between the expected agreement according to (multi)-π and the expected agreement according to (multi)-κ.[4] The bias is a measure of variance: if we take c to be a random variable with equal probabilities for all coders, then the bias B is the sum of the variances of $\hat{P}(k|c)$ for all categories $k \in K$, divided by the number of coders $\mathbf{c}$ less one (for proof see appendix A).

$$B = A_e^\pi - A_e^\kappa = \frac{1}{\mathbf{c} - 1} \sum_{k \in K} \sigma^2_{\hat{P}(k|c)}$$

This measure of bias can be used to express the difference between κ and π.

$$\kappa - \pi = \frac{A_o - (A_e^\pi - B)}{1 - (A_e^\pi - B)} - \frac{A_o - A_e^\pi}{1 - A_e^\pi} = B \cdot \frac{(1 - A_o)}{(1 - A_e^\kappa)(1 - A_e^\pi)}$$

This allows us to make the following observations about the relationship between π and κ.

1. For any particular coding data, $A_e^\pi \geq A_e^\kappa$, because B is the sum of non-negative numbers.
2. For any particular coding data, $\kappa \geq \pi$, because the difference between them is the product of non-negative numbers.
3. The difference between κ and π grows as the bias grows: for a constant A_o and A_e^π, a greater B implies a greater value for $\kappa - \pi$.

When bias is low, π and κ are similar and it doesn't make much of a difference which coefficient is used to measure reliability. A high bias is a likely indicator of a methodological flaw in the coding process, and is a source of disagreement in its own right.

It is also easy to show that the following holds:

Observation. *The greater the number of coders, the lower the bias* B*, and hence the lower the difference between κ and π, because the variance of $\hat{P}(k|c)$ does not increase in proportion to the number of coders.*

In other words, provided enough coders are used, it should not matter whether a biased or unbiased coefficient is used. This is not to imply that multiple coders increase reliability: the variance of the individual coders' distributions can be just as large with many coders as with few coders, but its effect on the value of κ decreases as the number of coders grows, and becomes more similar to random noise.

The coefficient β is related to α' in the same way that κ relates to π, namely: for any particular coding data, $D_e^\beta \geq D_e^{\alpha'}$ and consequently $\beta \geq \alpha'$, and the greater the number of coders, the lower the difference between β and α' (for proof see appendix B). We have already seen that α' and Krippendorff's α approach each other as either the number of items or the number of coders grows (section 2.5). This means that the more coders we have, the less important the choice of coefficient

[4] Our bias B is different from the Bias Index BI of Byrt et al. (1993).

among α, α', and β. In a recent annotation study with 18 subjects (Poesio and Artstein 2005) we found that in any particular condition, the values of α, α', and β did not differ beyond the third decimal point: for example, we found $\alpha = .69115$, $\alpha' = .69091$, and $\beta = .69111$ for the condition of full chains with Dice distance metric (see section 4.1 for an explanation of the various conditions).

4 Agreement coefficients for CL: The case for alpha (or beta)

Although most of the examples that Carletta used to motivate her choice of K came from work on segmentation, in practice it was soon found that κ is not very appropriate for this purpose (Marcu et al. 1999); problems were also encountered with other annotation tasks, such as anaphora / coreference. As a result, the κ coefficient has been primarily used in NLP to measure agreement on tasks involving tags with little or no possibility of overlap, such as dialogue act tagging (Carletta et al. 1997). However, Passonneau's recent proposals to use α for measuring agreement on anaphora annotation (Passonneau 2004) suggest that weighted measures may be a solution to many of these problems. In this section, we will discuss Passonneau's proposal, and list a variety of CL annotation tasks for which in our opinion weighted measures like α or β (or possibly weighted κ) are more appropriate than basic κ.

4.1 Measuring agreement on anaphoric annotation

Most authors who attempted anaphora annotation pointed out that κ is not appropriate for this task (Poesio and Vieira 1998; Byron 2003; Poesio 2004b; Passonneau 2004). The problem is that agreement between the most natural 'labels' for this task is not properly captured by an all-or-nothing measure.

In the case of anaphoric annotation it wouldn't make sense to predefine a set of 'labels' applicable to all texts, since different objects are mentioned in different texts. One possibility would be to use the marked antecedents as 'labels'. However, we do not want to count as a disagreement every case in which two coders agree on the discourse entity realized by a particular noun phrase (i.e., the agree about the anaphoric chain to which that mention belongs) but just happen to mark different members of that anaphoric chain as antecedents. Consider the dialogue excerpt in (1):[5] some of the coders in a study we recently carried out (Poesio and Artstein 2005) indicated *engine E2* as antecedent for the second *it* in utterance 3.1, whereas others indicated the immediately previous pronoun, which they had previously marked as having *engine E2* as antecedent. Clearly, we do not want to consider these coders to be in disagreement.

(1)
```
1.1  M: ....
1.4     first thing I'd like you to do
1.5     is send engine E2 off with a boxcar to Corning to
        pick up oranges
1.6     as soon as possible
2.1  S: okay
3.1  M: and while it's there it should pick up the tanker
```

[5] This example is taken from the first edition of the TRAINS corpus collected at the University of Rochester (Gross et al. 1993). The dialogues are available at ftp://ftp.cs.rochester.edu/pub/papers/ai/92.tn1.trains_91_dialogues.txt.

The most reasonable solution made in the literature is to use as 'labels' the *sets* of mentions of discourse entities, that is, anaphoric / coreference chains (Passonneau 2004). This proposal is in line with the methods developed to evaluate anaphora resolution systems (Vilain et al. 1995). But even using anaphoric chains as labels would not make κ a good measure for agreement. This is because κ only offers a dichotomous distinction between agreement and disagreement, whereas practical experience with anaphoric annotation suggests that except when a text is very short, few annotators will catch all mentions of a discourse entity: most will forget to mark a few, with the result that agreement as measured with κ is always very low. What is needed is a coefficient that also allows for partial disagreement between judgments, when two annotators agree on part of the coreference chain but not on all of it.

Passonneau (2004) suggests using α for this purpose, since this does allow for partial agreement among anaphoric chains. Passonneau (2004) proposes a distance metric between anaphoric chains based on the following rationale: two sets are minimally distant when they are identical and maximally distant when they are disjoint; between these extremes, sets that stand in a subset relation are closer (less distant) than ones that merely intersect. This leads to the following distance metric between two sets A and B.

$$\mathbf{d}_{AB} = \begin{cases} 0 & \text{if } A = B \\ 1/3 & \text{if } A \subset B \text{ or } B \subset A \\ 2/3 & \text{if } A \cap B \neq \emptyset, \text{ but } A \not\subset B \text{ and } B \not\subset A \\ 1 & \text{if } A \cap B = \emptyset \end{cases}$$

In the experiment just mentioned (Poesio and Artstein 2005), which used 18 coders, we tested α and β using both Passonneau's distance metric and distance metrics that take the size of the anaphoric chain into account, based on measures used to compare sets in Information Retrieval such as the coefficient of community of Jaccard (1912) and the coincidence index of Dice (1945) (Manning and Schuetze 1999). We found that even though our coders by and large agreed on the interpretation of anaphoric expressions, virtually no coder ever identified all the mentions of a discourse entity. As a result, even though the values of β and κ obtained by using the ID of the antecedent as label were pretty similar, the values obtained when using anaphoric chains as labels were drastically different. The value of β increased, because examples like (1) would no longer be considered as disagreements. However, the value of κ was drastically reduced, because hardly any coder identified all the mentions of discourse entities (see figure 2).

Chain	κ	β
None	0.62792	0.65549
Partial	0.56346	0.67662
Full	0.48108	0.69111

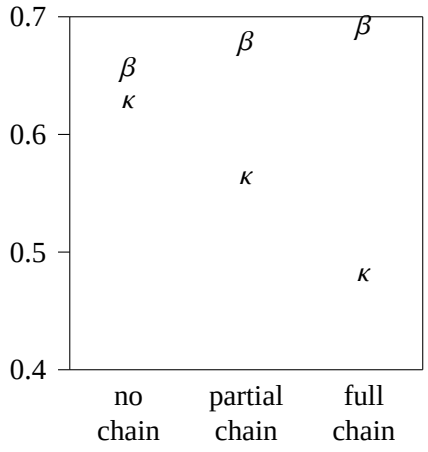

Figure 2: A comparison of the values of β and κ for anaphoric annotation

As we mentioned at the end of section 3, the study did not find differences between α, α', and β beyond the third decimal point. This similarity between biased and unbiased measures is what one would expect, given the result about bias from the previous section and given that in this experiment we used 18 annotators. These very small differences should be contrasted with the differences resulting from the choice of distance metrics, where values for the full-chain condition range from $\beta = .64197$ using Jaccard as distance metric, to $\beta = .65376$ using Passonneau's metric, to the value just reported for Dice, $\beta = .69111$. These differences raise an important issue concerning the application of α-like measures for CL tasks: using α is going to make it even more difficult to compare the results of different annotation experiments, in that a 'poor' value or a 'high' value might result from 'too strict' or 'too generous' distance metrics, making it even more important to develop a methodology to identify appropriate values for these coefficients.

4.2 Discourse Deixis

The annotation of **discourse deixis** (Webber 1991) gives rise to a different type of problem. Consider example (2), also from the TRAINS corpus.

(2)
```
3.1  M: and while it's there
3.2     it should pick up the tanker
4.1  S: okay
4.2     and that can get
4.3     we can get that done by three
```

Expressions like the demonstrative *that* in utterance 4.3 are generally interpreted as referring to some action, or plan, which is introduced by the previous discourse, but that cannot really be said to be the interpretation of a single expression. In order to mark this type of anaphoric references, annotators are generally required to mark a chunk of text such as a clause, utterance, sentence, or even turn. Most coding schemes for this type of expressions tend to introduce fairly strict restrictions on what the annotators should do (Eckert and Strube 2000); but if we allow coders at least the freedom to mark one or more utterance units, again we find the need to take partial overlap into account when measuring their agreement. This is because in cases such as the example in (2), most annotators will include 3.2 in the antecedent of discourse deictic *that*, but some will also include 3.1. In our recent study, subjects were allowed to mark any set of words as a discourse antecedent, irrespective of any clause, utterance, or turn boundaries. We computed agreement between such discourse antecedents by treating them as sets of words, and using the same distance metrics for sets as described in the previous section (Passonneau, Jaccard, and Dice).

4.3 Summarization

Evaluating content selection in summarization is a difficult problem (Radev et al. 2003), for which no single 'gold standard' can exist. (Machine translation faces a similar problem, as do all tasks involving generation.) Even if we only consider the simpler task of comparing summaries obtained by extracting sentences from the original document without any rephrasing, it is extremely unlikely that any two summaries will include exactly the same sentences – indeed, Lin and Hovy (2002) report that human summarizers agree with their own previous summaries in only about 82% of the cases. Clearly, a measure taking into account partial agreement such as α or β is needed to measure the agreement between coders producing a summary. Indeed, α has already been used for this purpose in a study by Nenkova and Passonneau (2004). Nenkova and Passonneau developed a method for evaluating summaries based on comparing them with a list of hand-annotated **summarization content units**,

broadly corresponding to the 'minimal propositions' expressed by the text, and used α to evaluate agreement on SCUS.

4.4 Other annotation tasks for which a weighted measure might be more appropriate

A number of other NLP tasks resemble anaphora resolution in that they require a graded measure of agreement, rather than an all-or-nothing one like π or κ. In this section, we list a few annotation tasks with which we have not experimented or we have only begun to experiment, but for which we conjecture that weighted measures such as α or β might be more appropriate than κ.

Segmentation The problems just discussed with the annotation of discourse deixis are reminiscent of the problems encountered when trying to evaluate agreement on discourse segmentation (Passonneau and Litman 1993; Hearst 1997; Marcu et al. 1999). The problem for this task is that while annotators generally agree on the 'bulk' of segments, they tend to disagree on their boundaries (see, e.g., Hearst's comparison of the annotations produced by seven coders of the same text in Figure 5 of Hearst 1997, page 55). As a result, an all-or-nothing coefficient like κ would indicate very low agreement even among annotators whose segmentation were mostly similar. A weighted coefficient of agreement like α or β would produce values more in keeping with intuition, especially – we conjecture – if a set-based distance metric such as Dice or Jaccard were used.

Wordsense Tagging The three SENSEVAL initiatives, and especially SENSEVAL-2 and SENSEVAL-3, highlighted the difficulty of sense tagging with a fine-grained repertoire of senses such as that provided by WordNet (Fellbaum 1998), particularly for the case of verbs, which tend to be polysemous rather than homonymous (Palmer et al. to appear). Agreement tests for SENSEVAL-2 reported a percentage agreement for verb senses of around 70%. The proposed solution has been to develop coarser grained classification schemes by grouping together senses, either manually on the basis of linguistic generalizations (Buitelaar 1998; Palmer et al. to appear) or automatically using principal component analysis (Bruce and Wiebe 1998). Using their grouped senses, Palmer et al. achieved a percentage agreement among coders of 82%. Measuring agreement beyond chance would be desirable, for the reasons discussed in the rest of the paper; we suggest that a graded measure like α or β would make it possible to allow annotators to mark either one of the grouped senses or one of the more specific senses, while maintaining a degree of agreement. This could be done by associating with each sense an **extended sense**, a set $\mathbf{es}(s)$ including the sense itself and its grouped sense, and then using the distance metric defined by Passonneau to measure the distance between the two senses:

$$\mathbf{d}_{s_A,s_B} = \begin{cases} 0 & \text{if } \mathbf{es}(s_A) = \mathbf{es}(s_B) \\ 1/3 & \text{if } \mathbf{es}(s_A) \subset \mathbf{es}(s_B) \text{ or } \mathbf{es}(s_B) \subset \mathbf{es}(s_A) \\ 2/3 & \text{if } \mathbf{es}(s_A) \cap \mathbf{es}(s_B) \neq \emptyset, \text{ but } \mathbf{es}(s_A) \not\subset \mathbf{es}(s_B) \text{ and } \mathbf{es}(s_B) \not\subset \mathbf{es}(s_A) \\ 1 & \text{if } \mathbf{es}(s_A) \cap \mathbf{es}(s_B) = \emptyset \end{cases}$$

To illustrate how this approach could be used to measure (dis)agreement on wordsense annotation, consider the example of *call* discussed in Palmer et al. (to appear). *Call* has 28 fine-grained senses in WordNet 1.7. Palmer et al. propose that four of these senses (1, 3, 19, 22) can be grouped in a group they call Group 1 on the basis of subcategorization frame similarities. The senses in this group are illustrated in Table 7, from Palmer et al. (to appear, page 13).

Suppose now that two coders have to annotate the use of *call* in the following sentence:[6]

[6]This sentence is from the WSJ part of the Penn Treebank, section 02, text w0209.

SENSE N	DESCRIPTION	EXAMPLE	HYPERNYM
WN1	name, call	"They named their son David"	**LABEL**
WN3	call, give a quality	"She called her children lazy and ungrateful"	**LABEL**
WN19	call, consider	"I would not call her beautiful"	**SEE**
WN22	address, call	"Call me mister"	**ADDRESS**

Table 7: Group 1 of senses of *call* in Palmer et al. (to appear).

(3) This gene, **called** "gametocide," is carried into the plant by a virus that remains active for a few days.

The standard guidelines (in SENSEVAL, say) require coders to assign a WN sense to words. Under such guidelines, if coder A classifies the use of *called* in (3) as an instance of WN1, whereas coder B annotates it as an instance of WN3, we would find total disagreement ($\mathbf{d}_{k_a k_b} = 1$) which seems excessively harsh as the two senses are clearly related. However, by using the broader senses proposed by Palmer et al. in combination with a distance metric such as the one just proposed, more flexible and, we believe, more realistic assessments of the degree of agreement in situations such as this become possible. For instance, in case the reliability study had already been carried out under the standard SENSEVAL guidelines, the distance metric proposed above could be used to identify *post hoc* cases of partial agreement by adding to each WN sense its hypernyms according to the groupings proposed by Palmer et al. For example, A's annotation could be turned into a new set label {WN1,**LABEL**} and B's mark into the set {WN3,**LABEL**}, which would give in a distance $\mathbf{d} = 2/3$, indicating a degree of overlap. The method for computing agreement proposed here could could also be used to allow coders to choose either a more specific label or one of Palmer et al.'s superlabels. For example, suppose A sticks to WN1, but B decides to mark the use above using Palmer et al.'s **LABEL** category. Then we would still find a distance $\mathbf{d} = 1/3$.

4.5 Evaluation of System Performance

Chance-discounted agreement coefficients are often used to evaluate system performance, by comparing the classification it produces with that produced by human coders (Poesio and Vieira 1998). Just as in the case of agreement among coders, chance-discounted coefficients are perceived as giving a better picture of system performance, which takes the actual difficulty of the task into proper account. In addition, comparing the agreement between system and coders with the agreement between coders is perceived as a fairer evaluation for many discourse and semantic interpretation tasks, in that it doesn't implicitly compare system performance against a gold standard that humans themselves cannot achieve. However, 'atomic' coefficients such as κ and π can only be used to compare all-or-nothing classifications; hence for tasks such as coreference, a chance-corrected alternative to standard metrics such as the one developed by Vilain et al. (1995) can only be provided by weighted coefficients like α and β. Such measures would also be needed for tasks such as term extraction, relation extraction, and, possibly, parsing.

Term extraction, for example, involves identifying a list of words that refer to a term, and can thus be considered as a simplified form of parsing. However, in biomedical texts, when terms are often modified, identifying the parts of a noun phrase that belong to a term may not be trivial (Travers and Haas 2003). For example, in the case of the NP *ID-2 protein*, the entire NP identifies a term; whereas

in the case of *ID-2 receptors*, two terms should be identified: *ID-2* and *ID-2 receptors*. Using an appropriate distance metric – e.g., one based on Dice, or on bigrams – would make it possible to use α or β to measure agreement on this task between a system and the coders.

5 Other Issues

5.1 Missing Data

An assumption that underlies all the coefficients that we have discussed is that all the coders classify all the items. In practice, however, this is not always the case, either because of practical limitations on the experimental setup or because some of the coders fail to classify certain items, for whatever reason. When data points are missing, the coefficients need to be adjusted to minimize the loss.

No adjustment is needed for the unbiased coefficients (π, α') if different items are classified by different coders, as long as the number of coders per item is constant; this is because the assumption of a single probability distribution for all coders means that the coders are indistinguishable. However, it is important that each item is classified by the same number of coders: the distribution of judgments over categories is estimated by tallying all the judgments, and if some items received more judgments than others they would receive more weight, skewing the estimate. This in turn would lead to an incorrect calculation of the expected agreement and of the overall coefficient value.

One solution to this problem is to eliminate the data points for which some judgments are missing, in order to achieve a data set were all coders classify all items. This is probably the best practice when the total data loss would be small. For example, Fleiss (1971) reports a psychiatric study in which each subject (item) was diagnosed (coded) by between 6 and 10 psychiatrists (coders), and states that "randomly selected diagnoses were dropped to bring the number of assignments per patient down to six." There is no indication of how much data were lost in this pruning. In a recent annotation experiment (Poesio and Artstein 2005), we had a total of 151 items, and data points were missing for three of them; eliminating these items from the analysis provided a quick solution to the missing data problem.

An alternative to dropping additional data is to redefine the observed and expected agreement and disagreement so as to minimize the skewing of the coefficient values. We start with the observed agreement and disagreement. Let $\mathbf{n}_i$ stand for the number of judgments available for a particular item i. We redefine the (unweighted) agreement value for item i as the proportion of agreeing judgment pairs for i out of the total number of judgment pairs for i; the (weighted) disagreement value for item i is the mean distance between the pairs of judgments pertaining to it.

$$\begin{aligned} \mathrm{agr}_i &= \frac{1}{\binom{\mathbf{n}_i}{2}} \sum_{k \in K} \binom{\mathbf{n}_{ik}}{2} = \frac{1}{\mathbf{n}_i(\mathbf{n}_i - 1)} \sum_{k \in K} \mathbf{n}_{ik}(\mathbf{n}_{ik} - 1) \\ \mathrm{disagr}_i &= \frac{1}{\mathbf{n}_i(\mathbf{n}_i - 1)} \sum_{j=1}^{\mathbf{k}} \sum_{l=1}^{\mathbf{k}} \mathbf{n}_{ik_j} \mathbf{n}_{ik_l} \mathbf{d}_{k_j k_l} \end{aligned}$$

If an item i receives only one judgment, then agr_i and disagr_i turn out to be $\frac{0}{0}$, or undefined; indeed in such a case one cannot talk of agreeing or disagreeing pairs at all. The overall observed (dis)agreement is the mean of the (dis)agreement values of the individual items, but calculated only on the items for which such a value exists. Let I' be the set of items for which there are two or more judgments available, and let $\mathbf{i}'$ be the cardinality of this set; observed agreement and disagreement can then be

defined as follows.

$$A_o = \frac{1}{\mathbf{i}'} \sum_{i \in I'} \text{agr}_i \qquad D_o = \frac{1}{\mathbf{i}'} \sum_{i \in I'} \text{disagr}_i$$

The expected agreement and disagreement for the unbiased measures (π, α') assumes a single probability distribution of items among categories for all coders. In order to ensure an equal weight to all items, we calculate this distribution as a weighted proportion, where for each item i the number of k judgments $\mathbf{n}_{ik}$ is scaled by $\mathbf{n}_i$, the total number of judgments for i.

$$\hat{P}(k) = \frac{1}{\mathbf{i}} \sum_{i \in I} \frac{\mathbf{n}_{ik}}{\mathbf{n}_i}$$

This is a probability distribution, which sums up to one. The formulas for A_e^{π} and $D_e^{\alpha'}$ remain as before, but using this new probability distribution.

The expected agreement and disagreement for the biased measures (κ, β) assumes a separate probability distribution of items among categories for each coder: $\hat{P}(k|c)$, the probability of assigning an item to category k by coder c, is the number of such assignments $\mathbf{n}_{ck}$ divided by the total number of assignments by the coder, $\mathbf{n}_c$.

$$\hat{P}(k|c) = \frac{\mathbf{n}_{ck}}{\mathbf{n}_c}$$

Since each coder judges a different number of items, we need to calculate weighted means for the coder pairs, reflecting the relative probabilities that an arbitrary pair consists of two particular coders. The probability that when choosing an arbitrary coder we would choose a particular coder c is the number of judgments given by c divided by the total number of judgments.

$$\hat{P}(c) = \frac{\mathbf{n}_c}{\sum_{c \in C} \mathbf{n}_c}$$

The probability that when choosing an arbitrary pair of coders we would choose two particular coders c_m and c_n is the joint probability of making these choices independently, divided by the sum of this joint probability for all coder pairs.

$$\hat{P}(c_m, c_n) = \frac{\hat{P}(c_m)\hat{P}(c_n)}{\sum_{a=1}^{\mathbf{c}-1} \sum_{b=a+1}^{\mathbf{c}} \hat{P}(c_a)\hat{P}(c_b)} = \frac{2\hat{P}(c_m)\hat{P}(c_n)}{1 - \sum_{c \in C} \hat{P}(c)^2}$$

This leads to the following revised formulas for A_e^{κ} and D_e^{β}.

$$\begin{aligned} A_e^{\kappa} &= \sum_{k \in K} \sum_{m=1}^{\mathbf{c}-1} \sum_{n=m+1}^{\mathbf{c}} \hat{P}(k|c_m)\hat{P}(k|c_n)\hat{P}(c_m, c_n) \\ D_e^{\beta} &= \sum_{j=1}^{\mathbf{k}} \sum_{l=1}^{\mathbf{k}} \sum_{m=1}^{\mathbf{c}-1} \sum_{n=m+1}^{\mathbf{c}} \hat{P}(k_j|c_m)\hat{P}(k_l|c_n)\hat{P}(c_m, c_n)\mathbf{d}_{k_j k_l} \end{aligned}$$

Finally, we note that an undated handout on Klaus Krippendorff's web site[7] shows how to compute α when some of the coding judgments are missing. The observed disagreement for each item is calculated as the mean distance among available judgment pairs, as we have done above. The expected disagreement is likewise the mean distance over all available judgment pairs without regard to items, but with the singleton judgments excluded. Since the mathematical principles underlying the definition of expected disagreement for α are not clear to us, it is also not clear what the effects of the missing data points would be on the calculation described here, and whether the missing data would skew the calculation of expected agreement.

[7] http://www.asc.upenn.edu/usr/krippendorff/webreliability2.pdf, accessed September 5, 2005.

5.2 Interpreting the value of kappa-like coefficients

Cohen's κ, Scott's π, Fleiss' multi-π, and Davies and Fleiss' multi-κ all come with significance tests that allow us to decide whether a particular level of agreement is significant. Many CL researchers however share the intuition expressed perhaps most elegantly by Cohen (1960) in his original article:[8]

> ... to know merely that κ is beyond chance is trivial since one usually expects much more than this in the way of reliability in psychological measurement. (page 44)

In CL, as well, knowing that coders agree above chance is rarely considered sufficient, whether one's goal is to measure agreement on linguistic judgments or to ensure that one's coding scheme is likely to result in resources of adequate quality. Unfortunately, to this day, deciding what counts as an 'acceptable' level of agreement for these purposes is little more than a black art. In the area of medical diagnosis, the best known conventions concerning the interpretation to be given to a value of a kappa-like coefficient are those due to Landis and Koch (1977) and reported in Table 8. Many medical researchers feel that these conventions, similar to those used for correlation coefficients, where values above .4 are also generally considered adequate (Marion 2004), are appropriate.

Kappa Values	Strength of Agreement
< 0.00	Poor
0.00–0.20	Slight
0.21–0.40	Fair
0.41–0.60	Moderate
0.61–0.80	Substantial
0.81–1.00	Perfect

Table 8: Kappa values and strength of agreement according to Landis and Koch (1977).

In CL, however, most researchers follow conventions adopted in Content Analysis, as suggested by Carletta (1996, page 252): "content analysis researchers generally think of $K > .8$ as good reliability, with $.67 < K < .8$ allowing tentative conclusions to be drawn."[9] As a result, ever since Carletta's extremely influential article, CL researchers have attempted to develop schemes that achieve a reliability above the magical .8 threshold, or, failing that, the .67 level allowing for "tentative conclusions". A more recent textbook on Content Analysis, Neuendorf (2002), analyzes several proposals concerning 'acceptable' reliability, concluding that "reliability coefficients of .9 or greater would be acceptable to all, .8 or greater would be acceptable in most situations, and below that, there exists great disagreement."

Practical experience with using these coefficients in CL hasn't helped in settling the issue. Our own experience is more consistent with Neuendorf's position than with that of Landis and Koch: in assessing semantic judgments, both in our earlier work (Poesio and Vieira 1998) and in the more recent (Poesio and Artstein 2005) we found that substantial, but by no means perfect, agreement among coders resulted in values of κ or α around the .7 level. But we also found that, in general, only values above .8 ensured a reasonable quality annotation (Poesio 2004a) – although again, there were some exceptions. On the other hand, even the lower .67 level has often proved impossible to

[8] A very clear expression of the same feelings can also be found in Posner et al. (1990).

[9] This assertion is based on a passage in Krippendorff (1980, page 147); it should be noted that Krippendorff was discussing values of α, rather than K.

achieve in CL research, particularly on discourse (see, e.g., Hearst 1997; Poesio and Vieira 1998). In fact, it is doubtful that a single cutoff point is appropriate for all tasks (for recent discussion, see Craggs and McGee Wood in press). And unfortunately, our suggestion that weighted coefficients are more appropriate for many annotation tasks make the issue of deciding when the value of a coefficient indicate sufficient agreement even more complicated. Neither Krippendorff's original α nor our β come with significance tests, although Krippendorff (1980, page 147) implies that there does exist a significance test for α; and with weighted measures, the value of the coefficient greatly depends on the distance metric chosen, as we saw at the end of section 4.1.[10]

These problems are one the reasons for the development of alternative measures of agreement such as those discussed in section 6.2. Researchers such as Uebersax (in the website mentioned in footnote 2) assert that the use of kappa-like measures to quantify 'levels of agreement' is "a source of concern," and that "aside from the significant / non significant determination, ... kappa's magnitude ... should be disregarded". We will briefly discuss the alternative proposed by Uebersax in section 6.2.

6 Beyond kappa: Non-analytical measures

A number of agreement measures proposed in the last twenty years rely heavily on computer estimation of expected agreement. Although some of these proposals are extremely promising, particularly the work on latent class analysis, covering this work would go beyond the scope of this paper, and we are not aware of any use of such techniques in CL. We will therefore provide only a brief summary of these proposals in this section.

6.1 Aickin's alpha

Aickin (1990) offers a way to measure agreement between two coders using an explicit model with a parameter called 'alpha'; we will designate it with the symbol **a** in order to distinguish it from Krippendorff's α. The model is based on four assumptions, the first three of which are as follows.

1. The items are made up of two populations, one which is easy to classify and one which is hard to classify.

2. The coders always agree on the classification of the easy items.

3. The coders classify hard items at random, according to personal probability distributions.

These assumptions give rise to an attractive conceptualization of the reliability problem. From our experience, items really differ in difficulty, and while it is clearly an oversimplification to make a dichotomous distinction between items that are classified at 100% accuracy and others that are classified at chance level, this does capture a basic intuition about the source of disagreement. Moreover, given the above assumptions, the formula for **a** is similar to that of the coefficients of the kappa family.

$$\mathbf{a} = \frac{A_o - A_e^{hard}}{1 - A_e^{hard}}$$

[10] One important step towards making the use of kappa-like measures less of a black art has been achieved: the development by Donner and Eliasziw (1987) of a more general form of significance test, which can be used to test the likelihood of a different group of coders achieving an *arbitrary* level ρ of agreement, as opposed to simply an agreement above chance (i.e., above 0). This method can be used, for instance, to test the likelihood of a second group of coders achieving $\kappa > .8$. Unfortunately, the test provides no guarantee that the level of agreement reached will be 'adequate' for the task at hand.

Here A_e^{hard} denotes the expected agreement, as calculated for Cohen's κ, but only on the population of hard items. The advantage of this over the kappa-like coefficients is that chance agreement is calculated only on the items which are assumed indeed to be randomly classified.

The model has a multitude of parameters – the proportion of easy items **a**, a distribution of easy items among categories, and two probability distributions of the hard items (one for each coder). Aickin reduces the number of parameters by adding a fourth assumption which ties together the distributions of the easy and hard items.

4. Easy items are distributed among those category pairs that denote agreement in proportion to the distribution of the hard items on which the two coders agree.

This last assumption entails that if the two coders agree on the classification of an item, the probability that this item is easy is the same irrespective of the category to which it was assigned; this property is called **constant predictive probability**. Even with this addition, the number of parameters in the model can be fairly large, and therefore Aickin uses maximum-likelihood estimation to estimate these parameters.

Aickin reports the result of simulations which were conducted with identical distributions for the two coders. The simulations show that the value of **a**, estimated by maximum likelihood, tends to be higher than that of κ, except when the distributions are uniform, in which case the two coefficients yield similar values. This is no surprise, as it can be shown analytically that $\kappa \leq \mathbf{a}$ in any data set generated by the model for coders with identical distributions, with the limiting case obtaining only when the coders' distributions are uniform (we omit the proof for lack of space). The reason for this is the fact that assumption 4 introduces a *nonlinear* relation between the distribution of the easy items and that of the hard items. It is easy to show that if we replace this assumption with a linear one 4', then $\mathbf{a} = \kappa = \pi$ for any data set generated by the model (this proof too is omitted for brevity).

4' Easy items are distributed among those category pairs that denote agreement in proportion to the individual coders' distributions for the hard items (assuming that these are identical for the two coders).

The modified assumption 4' results in a model which we will call the **linear distribution model**. We conjecture that simulations of this model would find the maximum-likelihood estimator for **a** very similar to the value of κ whenever the coders' distributions are identical, at least in coding data with a substantial amount of agreement (the model cannot generate data with systematic disagreement, and will therefore be a poor model for such data; note that with systematic disagreement κ can dip below zero, while zero is by definition the lower bound for **a**).

We feel that the linearity assumption 4' is more plausible than Aickin's original assumption 4. A natural interpretation of the linearity assumption is that the coders' individual probability distributions for the hard items are determined by the distribution of the easy items; an interpretation of Aickin's assumption 4 along these lines would imply that the coders infer a probability distribution of the hard items based on the *square roots* of the proportions of the easy items, which implies that the coders are somehow aware of the method of calculating agreement by looking at their joint decisions.

We thus have an additional interpretation of κ and π when the coders' distrbutions are identical, namely the proportion of items which are easy to classify in a linear distribution model (not Aickin's constant predictive probability model).

Finally, we note that our linear distribution model loses the constant predictive probability property of Aickin's original model, namely that if the two coders agree on the classification of an item, the probability that the item is easy is the same irrespective of the category to which it was assigned. We

feel that this is no great loss. The chance of spurious classification of an item into a common category is higher than the chance of spurious classification into a rare category, and therefore agreement on the classification of an item into a common category should indeed be less indicative that this agreement is genuine. This intuition, which we feel is valid, is contrary to the constant predictive probability hypothesis.

6.2 Latent Class Analysis

Latent class analysis (Uebersax 1988; Uebersax and Grove 1990) is the term used to refer to the application of **latent class modelling** techniques to nominal data. (The term **latent trait analysis** is used for ordinal categories.) Latent class modelling techniques, of which perhaps the best known example is the familiar EM algorithm (Goodman 1974; Dempster et al. 1977), were developed to deal with classification tasks in which the pattern of results (say, the POS tags assigned to words) derives from the membership of the items that have to be classified to an (unknown) number of (unknown) categories (the latent classes). Uebersax and colleagues showed how latent class analysis methods can be used to analyze agreement, and argued that such methods solve many of the problems they find with kappa-like coefficients of agreement (Uebersax 1988; Uebersax and Grove 1990). Although space prevents a full discussion of these methods, we think they are worth mentioning, especially as a way of addressing the problem of interpreting the value of κ. We will follow the discussion in the two papers by Uebersax cited above.

The rationale for the application of latent class methods to evaluate agreement is explained by Uebersax by comparing bimodal annotation (the annotationg of items which may belong to one of two classes – black and white, say) by, say, three coders to a two-step process: 1. extract a black or white marble from an urn; 2. then extract a number of marbles equal to the number of coders (three in this example) from one of two bowls: say, from bowl 1 if the marble is black, from bowl 2 if the marble is white. The urn represents the 'real' distribution of black and white marbles (i.e., $P(WHITE)$), whereas the two bowls represent the probability of an item being classified as white by, say, two annotators given that it's actually white: $P(2|WHITE)$. The results of this two-step process – that is, the observed behavior of the coders – can now be viewed as an instance of the well-known noisy channel model. So, for example, we can use the probability $P(white|WHITE)$ to characterize the likelihood that a given marble will be classified as white by the coders provided that it is actually white. Notice that this probability distribution is unknown; however, it clearly depends on the observed agreement among coders – the more the coders agree with each other on classifying items, the more likely it is that, say, a white marble will be actually classified as white.

What this way of thinking about coder behavior gives us is a model characterizing agreement in terms of diagnostic accuracy. For the two-label case, and assuming that there is a 'category of interest' (white, say), we can formalize diagnostic accuracy in terms of the notions of **sensitivity** and **specificity**. Sensitivity is the probability $P(white|WHITE)$ seen above of an item being marked as belonging to the class of interest by any of the coders if it's actually white. (This measure is equivalent to Recall for the 'positive' case.) Specificity is the probability $p(black|BLACK)$ of correctly classifying the items belonging to the category in which we are not interested. The problem of deciding whether agreement between coders is adequate then becomes one of testing that, say, sensitivity is above the threshold we need for this particular annotation task – .99, for example. As Uebersax and colleagues argue, this form of evaluation is both easier to understand and more directly related to the goals of the annotation than requiring, say, that the value of κ is above .8.[11]

[11]There is no direct correlation between a high value of κ and high sensitivity. For instance, Kraemer (1979) discusses

The problem of estimating the required probabilities given the output of the noisy channel is standard in CL, and a number of solutions have been developed. Uebersax and colleagues propose using maximum likelihood estimation to get initial estimates from the observed frequency counts, and then the EM algorithm to get more accurate estimates. This has to be done for models assuming different numbers of latent classes (the method does not require the assumption that the number of latent classes corresponds to the number of classes specified by the coding scheme); then the best class can be chosen using goodness of fit tests such as χ^2.

We find latent class analysis methods very promising; in addition to the advantages already discussed, they appear to offer the opportunity to decide how many ratings per item would be needed to ensure a certain sensitivity, and a variety of ways of analyzing the behavior of individual coders. However, they are to our knowledge still untested for CL applications, especially as a way of evaluating agreement on annotation.[12]

7 Conclusions

The coefficients of agreement belonging to the kappa family are very popular in CL, but a number of confusions about them remain, and they are not always used appropriately. In this paper we continued the work of re-examining these coefficients begun by Di Eugenio and Glass (2004) and Passonneau (2004), further clarifying similarities and differences between them. Our primary objective was to bring to the attention of the community the fact that weighted coefficients such as α and weighted κ, recently championed by Passonneau and others, are not just notational variants of the more popular K discussed by Carletta, but are more powerful and, we believe, more appropriate for CL annotation tasks such as anaphora / coreference, segmentation, and wordsense. While doing so, we returned to the issue of bias raised by Di Eugenio and Glass. We believe that assuming that coders act in accordance with the same probability distribution is too strong of an assumption, hence 'biased' measures are more appropriate; consequently, we developed a 'biased' version of α, which we called β. However we also noted that bias doesn't make much difference in empirical practice, and we also showed analytically that the effect of bias disappears as the number of coders grows.

Interpreting the value of kappa-like coefficients remains in our view the main open problem. On the negative side, we noticed that using weighted coefficients makes this problem even more difficult. On the other hand, we also think that more recent proposals, in particular latent class analysis, may well be the solution to this problem.

A Bias and variance with multiple coders

In section 3 we briefly noted that the difference between π and κ drops as the number of coders increases. Here we give the formal proof. We start by taking the formulas for expected agreement from section 2.4 and putting them into a form that is more useful for comparison with one another.

$$A_e^\pi = \sum_{k \in K} \hat{P}(k)^2 = \sum_{k \in K} \left(\frac{1}{\mathbf{c}} \sum_{m=1}^{\mathbf{c}} \hat{P}(k|c_m) \right)^2$$

cases in which high sensitivity and specificity are found with κ=.4338.

[12] Bruce and Wiebe (1998) used latent class modelling to identify the number of latent classes that seemed to underly the behavior of their coders at the wordsense annotation task.

$$
\begin{aligned}
&= \sum_{k\in K} \frac{1}{\mathbf{c}^2} \sum_{m=1}^{\mathbf{c}} \sum_{n=1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m)\hat{\mathrm{P}}(k|c_n) \\
\mathrm{A}_\mathrm{e}^\kappa &= \sum_{k\in K} \frac{1}{\binom{\mathbf{c}}{2}} \sum_{m=1}^{\mathbf{c}-1} \sum_{n=m+1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m)\hat{\mathrm{P}}(k|c_n) \\
&= \sum_{k\in K} \frac{1}{\mathbf{c}(\mathbf{c}-1)} \left(\sum_{m=1}^{\mathbf{c}} \sum_{n=1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m)\hat{\mathrm{P}}(k|c_n) - \sum_{m=1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m)^2 \right)
\end{aligned}
$$

The overall bias B is the difference between the expected agreement according to π and the expected agreement according to κ.

$$
\begin{aligned}
\mathrm{B} &= \mathrm{A}_\mathrm{e}^\pi - \mathrm{A}_\mathrm{e}^\kappa \\
&= \frac{1}{\mathbf{c}-1} \sum_{k\in K} \frac{1}{\mathbf{c}^2} \left(\mathbf{c} \sum_{m=1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m)^2 - \sum_{m=1}^{\mathbf{c}} \sum_{n=1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m)\hat{\mathrm{P}}(k|c_n) \right)
\end{aligned}
$$

We now calculate the mean μ and variance σ^2 of $\hat{\mathrm{P}}(k|c)$, taking c to be a random variable with equal probabilities for all of the coders: $\hat{\mathrm{P}}(c) = \frac{1}{\mathbf{c}}$ for all coders $c \in C$.

$$
\begin{aligned}
\mu_{\hat{\mathrm{P}}(k|c)} &= \frac{1}{\mathbf{c}} \sum_{m=1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m) \\
\sigma^2_{\hat{\mathrm{P}}(k|c)} &= \frac{1}{\mathbf{c}} \sum_{m=1}^{\mathbf{c}} (\hat{\mathrm{P}}(k|c_m) - \mu_{\hat{\mathrm{P}}(k|c)})^2 \\
&= \frac{1}{\mathbf{c}} \sum_{m=1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m)^2 - 2\mu_{\hat{\mathrm{P}}(k|c)} \frac{1}{\mathbf{c}} \sum_{m=1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m) + \mu^2_{\hat{\mathrm{P}}(k|c)} \frac{1}{\mathbf{c}} \sum_{m=1}^{\mathbf{c}} 1 \\
&= \left(\frac{1}{\mathbf{c}} \sum_{m=1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m)^2 \right) - \mu^2_{\hat{\mathrm{P}}(k|c)} \\
&= \frac{1}{\mathbf{c}^2} \left(\mathbf{c} \sum_{m=1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m)^2 - \sum_{m=1}^{\mathbf{c}} \sum_{n=1}^{\mathbf{c}} \hat{\mathrm{P}}(k|c_m)\hat{\mathrm{P}}(k|c_n) \right)
\end{aligned}
$$

The bias B is thus the sum of the variances of $\hat{\mathrm{P}}(k|c)$ for all categories $k \in K$, divided by the number of coders less one.

$$\mathrm{B} = \frac{1}{\mathbf{c}-1} \sum_{k\in K} \sigma^2_{\hat{\mathrm{P}}(k|c)}$$

Since the variance does not increase in proportion to the number of coders, we find that the more coders we have, the lower the bias; at the limit, κ approaches π as the number of coders approaches infinity.

B Bias of weighted measures

We have shown in appendix A that the variance of the individual coders' distibutions of items to categories is a useful measure for the bias in a set of coding data, and that it correlates with the

difference between π and κ. This measure of variance is less useful when the coding data are judged according to a weighted measure, because the discrepancies between the individual coders also have varying magnitudes. A measure of bias for such coding data should therefore take the weights into account. Since the expected disagreement already considers the weights, we define the bias B in an analogous way to our definition in appendix A, namely as the difference between the expected disagreement according to the unbiased measure α' and the expected disagreement according to the biased measure β.

$$\mathrm{B} = \mathrm{D}_\mathrm{e}^{\alpha'} - \mathrm{D}_\mathrm{e}^{\beta}$$

We first put the expected disagreements according to α' and β (sections 2.5 and 2.7 respectively) into forms that are more useful for the comparison.

$$\begin{aligned}
\mathrm{D}_\mathrm{e}^{\alpha'} &= \sum_{j=1}^{\mathbf{k}}\sum_{l=1}^{\mathbf{k}} \hat{\mathrm{P}}(k_j)\hat{\mathrm{P}}(k_l)\mathbf{d}_{k_jk_l} = \sum_{j=1}^{\mathbf{k}}\sum_{l=1}^{\mathbf{k}} \frac{1}{\mathbf{c}^2}\sum_{m=1}^{\mathbf{c}}\sum_{n=1}^{\mathbf{c}} \hat{\mathrm{P}}(k_j|c_m)\hat{\mathrm{P}}(k_l|c_n)\mathbf{d}_{k_jk_l} \\
\mathrm{D}_\mathrm{e}^{\beta} &= \frac{1}{\binom{\mathbf{c}}{2}}\sum_{j=1}^{\mathbf{k}}\sum_{l=1}^{\mathbf{k}}\sum_{m=1}^{\mathbf{c}-1}\sum_{n=m+1}^{\mathbf{c}} \hat{\mathrm{P}}(k_j|c_m)\hat{\mathrm{P}}(k_l|c_n)\mathbf{d}_{k_jk_l} \\
&= \sum_{j=1}^{\mathbf{k}}\sum_{l=1}^{\mathbf{k}} \frac{1}{\mathbf{c}(\mathbf{c}-1)}\left(\sum_{m=1}^{\mathbf{c}}\sum_{n=1}^{\mathbf{c}} \hat{\mathrm{P}}(k_j|c_m)\hat{\mathrm{P}}(k_l|c_n) - \sum_{m=1}^{\mathbf{c}} \hat{\mathrm{P}}(k_j|c_m)\hat{\mathrm{P}}(k_l|c_m)\right)\mathbf{d}_{k_jk_l}
\end{aligned}$$

Now we calculate the bias as the difference between the above measures.

$$\begin{aligned}
\mathrm{B} &= \mathrm{D}_\mathrm{e}^{\alpha'} - \mathrm{D}_\mathrm{e}^{\beta} \\
&= \sum_{j=1}^{\mathbf{k}}\sum_{l=1}^{\mathbf{k}}\Bigg((\frac{1}{\mathbf{c}^2} - \frac{1}{\mathbf{c}(\mathbf{c}-1)})\sum_{m=1}^{\mathbf{c}}\sum_{n=1}^{\mathbf{c}} \hat{\mathrm{P}}(k_j|c_m)\hat{\mathrm{P}}(k_l|c_n) \\
&\qquad + \frac{1}{\mathbf{c}(\mathbf{c}-1)}\sum_{m=1}^{\mathbf{c}} \hat{\mathrm{P}}(k_j|c_m)\hat{\mathrm{P}}(k_l|c_m)\Bigg)\mathbf{d}_{k_jk_l} \\
&= \frac{1}{\mathbf{c}-1}\sum_{j=1}^{\mathbf{k}}\sum_{l=1}^{\mathbf{k}} \frac{1}{\mathbf{c}^2}\left(\mathbf{c}\sum_{m=1}^{\mathbf{c}} \hat{\mathrm{P}}(k_j|c_m)\hat{\mathrm{P}}(k_l|c_m) - \sum_{m=1}^{\mathbf{c}}\sum_{n=1}^{\mathbf{c}} \hat{\mathrm{P}}(k_j|c_m)\hat{\mathrm{P}}(k_l|c_n)\right)\mathbf{d}_{k_jk_l}
\end{aligned}$$

Unlike the bias for unweighted measures, this measure of bias does not correspond to the sum of the variances of a single random variable But the bias still drops in proportion to an increase in the number of coders: the sums inside the parentheses grow in proportion to $\mathbf{c}^2$, and therefore the overall bias B grows in proportion to $\frac{1}{\mathbf{c}-1}$.

Acknowledgments

This work was in part supported by EPSRC grant GR/S76434/01, ARRAU. We wish to thank Barbara Di Eugenio, Ruth Filik, Michael Glass, Becky Passonneau, Tony Sanford, and Patrick Sturt for helpful comments and discussion. We are also extremely grateful to the British Library in London, which made accessible to us virtually every paper we needed for this research.

References

Aickin, Mikel (1990). Maximum likelihood estimation of agreement in the constant predictive probability model, and its relation to Cohen's kappa. *Biometrics*, **46**(2):293–302.

Allen, James and Mark Core (1997). DAMSL: Dialogue act markup in several layers. http://www.cs.rochester.edu/research/cisd/resources/damsl/. Draft contribution for the Discourse Resource Initiative.

Bartko, John J. and William T. Carpenter, Jr (1976). On the methods and theory of reliability. *Journal of Nervous and Mental Disease*, **163**(5):307–317.

Bennett, E. M., R. Alpert, and A. C. Goldstein (1954). Communications through limited questioning. *Public Opinion Quarterly*, **18**(3):303–308.

Brennan, Robert L. and Dale J. Prediger (1981). Coefficient kappa: Some uses, misuses, and alternatives. *Educational and Psychological Measurement*, **41**(3):687–699.

Bruce, Rebecca and Janyce Wiebe (1998). Word-sense distinguishability and inter-coder agreement. In *Proceedings of EMNLP*, pages 53–60. Granada, Spain.

Buitelaar, Paul (1998). *CoreLex : Systematic Polysemy and Underspecification*. Ph.D. thesis, Brandeis University.

Byron, Donna K. (2003). Annotation of pronouns and their antecedents: A comparison of two domains. Technical Report 703, University of Rochester Computer Science Department. ftp://ftp.cs.rochester.edu/pub/papers/ai/03.tr703.Annotation_of_pronouns_and_their_antecedents.ps.gz.

Byrt, Ted, Janet Bishop, and John B. Carlin (1993). Bias, prevalence and kappa. *Journal of Clinical Epidemiology*, **46**(5):423–429.

Carletta, Jean (1996). Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, **22**(2):249–254.

Carletta, Jean, Amy Isard, Stephen Isard, Jacqueline C. Kowtko, Gwyneth Doherty-Sneddon, and Anne H. Anderson (1997). The reliability of a dialogue structure coding scheme. *Computational Linguistics*, **23**(1):13–32.

Cicchetti, Domenic V. and Alvan R. Feinstein (1990). High agreement but low kappa: II. Resolving the paradoxes. *Journal of Clinical Epidemiology*, **43**(6):551–558.

Cohen, Jacob (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, **20**(1):37–46.

Cohen, Jacob (1968). Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological Bulletin*, **70**(4):213–220.

Craggs, Richard and Mary McGee Wood (in press). Evaluating discourse and dialogue coding schemes. *Computational Linguistics*.

Davies, Mark and Joseph L. Fleiss (1982). Measuring agreement for multinomial data. *Biometrics*, **38**(4):1047–1051.

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, **39**(1):1–38.

Di Eugenio, Barbara (2000). On the usage of Kappa to evaluate agreement on coding tasks. In *Proceedings of LREC*, volume 1, pages 441–444. Athens.

Di Eugenio, Barbara and Michael Glass (2004). The kappa statistic: A second look. *Computational Linguistics*, **30**(1):95–101.

Dice, Lee R. (1945). Measures of the amount of ecologic association between species. *Ecology*, **26**(3):297–302.

Donner, Allan and Michael Eliasziw (1987). Sample size requirements for reliability studies. *Statistics*

in Medicine, **6**:441–448.

Eckert, Miriam and Michael Strube (2000). Dialogue acts, synchronizing units, and anaphora resolution. *Journal of Semantics*, **17**(1):51–89.

Feinstein, Alvan R. and Domenic V. Cicchetti (1990). High agreement but low kappa: I. The problems of two paradoxes. *Journal of Clinical Epidemiology*, **43**(6):543–549.

Fellbaum, Christiane (editor) (1998). *WordNet: An Electronic Lexical Database*. MIT Press.

Fellbaum, Christiane, Joachim Grabowski, and Shari Landes (1997). Analysis of a hand-tagging task. In *Proceedings of ANLP Workshop on Tagging Text with Lexical Semantics*, pages 34–40. Washington, DC.

Fleiss, Joseph L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, **76**(5):378–382.

Francis, W. Nelson and Henry Kucera (1982). *Frequency Analysis of English Usage: lexicon and grammar*. Houghton Mifflin, Boston.

Goodman, Leo A. (1974). The analysis of systems of qualitative variables when some of the variables are unobservable. Part I – a modified latent structure approach. *American Journal of Sociology*, **79**(5):1179–1259.

Gross, Derek, James F. Allen, and David R. Traum (1993). The Trains 91 dialogues. TRAINS Technical Note 92-1, University of Rochester Computer Science Department. ftp://ftp.cs.rochester.edu/pub/papers/ai/92.tn1.trains_91_dialogues.ps.Z.

Hearst, Marti A. (1997). TextTiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, **23**(1):33–64.

Hsu, Louis M. and Ronald Field (2003). Interrater agreement measures: Comments on kappa$_n$, Cohen's kappa, Scott's π, and Aickin's α. *Understanding Statistics*, **2**(3):205–219.

Jaccard, Paul (1912). The distribution of the flora in the Alpine zone. *New Phytologist*, **11**(2):37–50.

Kraemer, Helena Chmura (1979). Ramifications of a population model for κ as a coefficient of reliability. *Psychometrika*, **44**(4):461–472.

Krippendorff, Klaus (1970). Bivariate agreement coefficients for reliability of data. *Sociological Methodology*, **2**:139–150.

Krippendorff, Klaus (1980). *Content Analysis: An Introduction to Its Methodology*, chapter 12, pages 129–154. Sage, Beverly Hills.

Landis, J. Richard and Gary G. Koch (1977). The measurement of observer agreement for categorical data. *Biometrics*, **33**(1):159–174.

Lin, Chin-Yew and Eduard Hovy (2002). Manual and automatic evaluation of summaries. In *Proceedings of the ACL-02 Workshop on Automatic Summarization*, pages 45–51. Philadelphia.

Manning, Christopher D. and Hinrich Schuetze (1999). *Foundations of Statistical Natural Language Processing*. MIT Press.

Marcu, Daniel, Magdalena Romera, and Estibaliz Amorrortu (1999). Experiments in constructing a corpus of discourse trees: Problems, annotation choices, issues. In *Workshop on Levels of Representation in Discourse*, pages 71–78. University of Edinburgh.

Marion, Rodger (2004). The whole art of deduction. http://sahs.utmb.edu/pellinore/intro_to_research/wad/correlat.htm.

Nenkova, Ani and Rebecca Passonneau (2004). Evaluating content selection in summarization: The pyramid method. In Susan Dumais, Daniel Marcu, and Salim Roukos (editors), *Proceedings of HLT-NAACL 2004*, pages 145–152. Association for Computational Linguistics, Boston.

Neuendorf, Kimberly A. (2002). *The Content Analysis Guidebook*. Sage, Thousand Oaks, CA.

Palmer, Martha, Hoa Trang Dang, and Christiane Fellbaum (to appear). Making fine-grained and coarse-grained sense distinctions, both manually and automatically. *Natural Language Engineer-*

ing.

Passonneau, Rebecca J. (2004). Computing reliability for coreference annotation. In *Proceedings of LREC*, volume 4, pages 1503–1506. Lisbon.

Passonneau, Rebecca J. and Diane J. Litman (1993). Intention-based segmentation: Human reliability and correlation with linguistic cues. In *Proceedings of 31st Annual Meeting of the ACL*, pages 148–155. Columbus, OH.

Poesio, Massimo (2004a). Discourse annotation and semantic annotation in the GNOME corpus. In *Proceedings of the ACL Workshop on Discourse Annotation*. Barcelona.

Poesio, Massimo (2004b). The MATE/GNOME proposals for anaphoric annotation, revisited. In Michael Strube and Candy Sidner (editors), *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue*, pages 154–162. Cambridge, MA.

Poesio, Massimo and Ron Artstein (2005). The reliability of anaphoric annotation, reconsidered: Taking ambiguity into account. In *Proceedings of the Workshop on Frontiers in Corpus Annotation II: Pie in the Sky*, pages 76–83. Association for Computational Linguistics, Ann Arbor, Michigan. http://www.aclweb.org/anthology/W/W05/W05-0311.

Poesio, Massimo and Renata Vieira (1998). A corpus-based investigation of definite description use. *Computational Linguistics*, **24**(2):183–216.

Posner, Karen L., Paul D. Sampson, Robert A. Caplan, Richard J. Ward, and Frederick W. Cheney (1990). Measuring interrater reliability among multiple raters: an example of methods for nominal data. *Statistics in Medicine*, **9**:1103–1115.

Radev, Dragomir R., Simone Teufel, Horacio Saggion, Wai Lam, John Blitzer, Hong Qi, Arda Çelebi, Danyu Liu, and Elliott Drabek (2003). Evaluation challenges in large-scale document summarization. In *Proceedings of 41st Annual Meeting of the ACL*, pages 375–382. Sapporo.

Scott, William A. (1955). Reliability of content analysis: The case of nominal scale coding. *Public Opinion Quarterly*, **19**(3):321–325.

Siegel, Sidney and N. John Castellan, Jr (1988). *Nonparametric Statistics for the Behavioral Sciences*, chapter 9.8, pages 284–291. 2nd edition. McGraw-Hill, New York.

Stevenson, Mark and Robert Gaizauskas (2000). Experiments on sentence boundary detection. In *Proceedings of 6th ANLP*, pages 84–89. Seattle.

Stuart, Alan (1955). A test for homogeneity of the marginal distributions in a two-way classification. *Biometrika*, **42**(3/4):412–416.

Travers, Debbie A. and Stephanie W. Haas (2003). Using nurses' natural language entries to build a concept-oriented terminology of patients' chief complaints in the emergency department. *Journal of Biomedical Informatics*, **36**(4/5):260–270.

Uebersax, John S. (1988). Validity inferences from interobserver agreement. *Psychological Bulletin*, **104**(3):405–416.

Uebersax, John S. and William M. Grove (1990). Latent class analysis of diagnostic agreement. *Statistics in Medicine*, **9**:559–572.

Vilain, Marc, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman (1995). A model-theoretic coreference scoring scheme. In *Proceedings of the Sixth Message Understanding Conference*, pages 45–52. Columbia, Maryland.

Webber, Bonnie Lynn (1991). Structure and ostension in the interpretation of discourse deixis. *Language and Cognitive Processes*, **6**(2):107–135.

Zwick, Rebecca (1988). Another look at interrater agreement. *Psychological Bulletin*, **103**(3):374–378.

Other Technical Notes and theses from the Natural Language Engineering and Web Applications group are available electronically at

http://cswww.essex.ac.uk/Research/nle/

Dept. of Computer Science
University of Essex
Wivenhoe Park
Colchester CO4 3SQ
United Kingdom