On the usage of Kappa to evaluate agreement on coding tasks

Barbara Di Eugenio

Electrical Engineering and Computer Science
University of Illinois at Chicago
Chicago, IL 60607, USA
dieugeni@eecs.uic.edu

Abstract

In recent years, the Kappa coefficient of agreement has become the de facto standard to evaluate intercoder agreement in the discourse and dialogue processing community. Together with the adoption of this standard, researchers have adopted one specific scale to evaluate Kappa values, the one proposed in (Krippendorff, 1980). In this position paper, I highlight some issues that should be taken into account when evaluating Kappa values. Finally, I speculate on whether Kappa could be used as a measure to evaluate a system's performance.

1. Introduction

In the last few years, coded corpora have acquired an increasing importance in almost every aspect of human language technology, from part-of-speech tagging to discourse and dialogue processing. Apart form part-of-speech tagging where semi-automatic techniques for tagging have been very successful, tagging for all other phenomena of interest (from syntactic annotations to anaphoric links to dialogue acts) is still mainly a manual effort. This raises the question of how to evaluate the "goodness" of a coding scheme. One way of doing so is to assess its reliability, namely, to assess whether different coders can reach a satisfying level of agreement with each other when they use the coding manual on the same data.

In the discourse and dialogue processing community, until about five years ago, agreement was measured as the percentage of the cases on which coders agree. Now, the de facto standard is the Kappa coefficient of agreement that factors out expected agreement (Cohen, 1960; Krippendorff, 1980). Carletta is the researcher who brought this measure to the attention of the discourse and dialogue processing community. In (Carletta, 1996), she convincingly argued that Kappa should be used, because the percentage of times two coders agree with each other is not a meaningful measure, as it is obfuscated by chance agreement. For example, if two categories occur in equal proportions, coders would agree with each other by chance half of the time.

In this paper, I discuss some issues that should be taken into account when using Kappa. Moreover, I suggest one way in which Kappa could be used as an additional way to evaluate the performance of the system that is trained on the tagged data.

2. The Kappa coefficient of agreement

The Kappa coefficient of agreement (Cohen, 1960; Krippendorff, 1980)[1] measure of agreement that factors out expected agreement. Kappa has been long used in content analysis and medicine to assess the reliability of tagging (for example, in medicine, to assess how well students' diagnoses on a set of test cases agree with expert answers). In the formula in (1), $P(A)$ is observed agreement, and $P(E)$ is expected agreement.

(1)
$$K = \frac{P(A) - P(E)}{1 - P(E)}$$

Kappa's possible values are constrained to the interval $[0, 1]$; K=0 means that agreement is not different from chance, and K=1 means perfect agreement. However, just obtaining a K significantly greater than zero is not sufficient to assess the "quality" of the agreement. Various scales to assess Kappa's significance have been proposed, the strictest one being Krippendorff's (Krippendorff, 1980): this scale discounts any variable with $K < .67$, allows tentative conclusions when $.67 < K < .8$ K, and definite conclusions when $K \geq .8$. There are other more forgiving scales, e.g., (Rietveld and van Hout, 1993) consider $.41 < K < .60$ as indicating moderate agreement, and $.61 < K < .80$ as indicating substantial agreement. The psychiatric community considers $K > .6$ or even $K > .5$ as acceptable (Grove et al., 1981).

Without any real assessment, the discourse processing community has adopted Krippendorff's scale for assessing intercoder reliability. This scale has been adopted without questions even if Krippendorff himself reports this scale only as a plausible standard that has emerged from his and his colleagues' work. He also states that the significance of any such standard cannot be absolutely stated, but depends on the usage of the results that one derives from the analysis, and in particular, on the cost of wrong conclusions (Krippendorff, 1980, ch. 12).

In my opinion, the dialogue and discourse processing community should pay more attention to the meaning of the scales used to evaluate Kappa values. Part of the scientific value of a coding scheme is now assessed on the basis of Kappa values, and coded data becomes the basis for data mining from text, and for system implementation.

2.1. Factors that affect Kappa values

Different factors affect Kappa in different ways. Some factors affect the possible values of Kappa per se, because

[1] Although Krippendorff proposes the coefficient α as an extension to Kappa, for nominal scales and for two coders the two measures are equivalent (Carletta, 1996; Passonneau, 1997).

they affect its computation; other factors only affect the interpretation of those values. In the following, I will discuss two factors that affect the computation of Kappa, the computation of the expected agreement P(E) and the distribution of categories; and one factor that affects the interpretation of Kappa values.

2.1.1. Computing Kappa

Computing P(E). There are differences in the way P(E), the expected agreement, is calculated. They correspond to whether the distribution of proportions over the categories is taken to be equal for the judges (Siegel and Castellan, 1988) or not (Cohen, 1960; Krippendorff, 1980; Passonneau, 1997; Wiebe et al., 1999). Thus, adopting one or the other measure affects the values of Kappa, and in turn, should be taken into account when assessing them.

Skewed distribution of categories. In previous work (Di Eugenio et al., 1998; Di Eugenio et al., 2000), we reported the results of an extensive coding effort we undertook. We collected 24 computer-mediated design dialogues in which two people collaborate on a simple design task, buying furniture for the living and dining rooms of a house. 9 of the 24 dialogues were doubly annotated by 2 annotators, for a total of 482 coded utterances. We coded for two aspects of the conversations we collected: the dialogue features proper, and the domain reasoning situation. We designed the part of our coding scheme concerning the dialogue to conform with the standards that were being developed within the Discourse Resource Initiative (DRI)[2] DRI produced a draft annotation scheme called DAMSL (DAMSL, 1997).

Two dimensions we coded for that I will discuss in this paper are: *Forward-Looking Functions*, that characterize the effect that utterance U_i has on the subsequent dialogue, and that roughly correspond to the classical notion of an *illocutionary act* (Austin, 1962; Searle, 1965; Searle, 1975); and *Backward-Looking Functions*, that indicate whether U_i is unsolicited, or provides a response of some sort to a previous U_j or segment.

Forward-Looking Functions and *Backward-Looking Functions* are further specialized. Regarding the former dimension, each U_i may be coded along one or more of the four different subdimensions: *Statement*, *Influence-on-Hearer*, *Influence-on-Speaker*, *Other-Forward-Function*. Briefly, the primary purpose of *Statements* is "to make claims about the world". U_i tagged along the *Influence-on-Hearer* dimension is intended to influence the hearer's future actions, whereas a U_i tagged along the *Influence-on-Speaker* dimension potentially commits the speaker to some future course of action. *Influence-on-Hearer* tags include *Open-Option* (the speaker is merely laying out options for the hearer's future actions), *Action-Directives* (the speaker is putting the hearer under obligation to act (**?**)), and *Info-Request* (includes all actions that request information). Finally, *Other-forward-function* include conventional conversational acts such as greetings, explicit performatives, and exclamations.

As regards *Backward-Looking Functions*, the ones more relevant to the current discussion are as follows. *Answer* is used when U_i answers a question. *Agreement* tags are used when U_i expresses S's attitude towards a belief or option for action embodied in its antecedent. Agreement tags include *Accept, Reject* and *Hold*, used when U_i does not express an attitude towards its antecedent, but leaves the decision open pending further discussion.

Tables 1 presents the Kappa results for Forward and Backward looking functions (all of our K values are significant at p=0.000005, except for *Other-forward-function* at p=0.0005). We also coded for a variety of other features, such as *Gist* tags, that capture the gist of the utterance in terms of features relevant to problem solving; *Reference* tags, that encode a simple notion of reference relations; syntactic properties of the utterance. We obtained values of Kappa greater than .8 for all these supplementary tags.

The columns in the tables read as follows: is utterance U_i tagged for tag *X*, and if yes, do coders agree on the specific subtag? For example, the possible set of values for *Influence-on-Listener* are: NIL (U_i is not tagged along this dimension), Action-Directive, Open-Option, and Info-Request. The last two columns probe Backward Functions: was U_i tagged as an answer? was U_i tagged as *accept*ing, *reject*ing, or *hold*ing the same antecedent? Computing Kappa for the backward tags takes into account whether the coders linked U_i to the same antecedent: thus, a situation in which both coders code U_i as *Accept*, but disagree on what antecedent U_i accepts, counts as a disagreement.

Whatever scale one adopts, Table 1 suggests that Forward Functions and Answers can be recognized far more reliably than Agreement functions. The question we asked ourselves is: why is the Kappa value on Agreement tags so unsatisfactory? One possible explanation is that agreement tags are much rarer than Forward Function tags, rather than to a basic flaw in the definition of agreement. Out of the 482 utterances in Table 1, in one coder's tagged data there are only 75 occurrences of an agreement tag, and in the other coder's, only 46. This pushes the expected agreement up (because coders agree most of the time simply by not tagging U_i for agreement), thus a very high level of agreement on the tags that do occur is necessary to reach good results. This intuitive explanation is backed up by (Grove et al., 1981), which points out that the low frequency of a tag may lower the maximum K (corresponding to perfect agreement) to a value sometimes much lower than 1. On this topic, see also the exchange between (Berry, 1992) and (Goldman, 1992). Whether the argument in (Grove et al., 1981) can be formally applied to discourse and dialogue processing work is not clear, because (Grove et al., 1981) makes use of a measure of *validity*, i.e., of a gold standard against which the coders' analyses can be assessed, which is not available yet in discourse processing work.

2.1.2. Interpreting Kappa values

Tagging in content analysis or in medicine generally consists of assigning one judgement per case, such as whether an article expresses support for the Chinese government (Krippendorff, 1980), or whether a patient in a case study is classified as schizophrenic (Grove et al., 1981). However, tagging for discourse/dialogue ofter calls for tagging for categories that are not independent. That is,

[2] See http://www.georgetown.edu/luperfoy/Discourse-Treebank/dri-home.html.

Forward Functions				Backward Functions	
Statement	Listener	Speaker	Other	Answer	Agreement
.83	.72	.72	.93	.79	.54

Table 1: Kappa values for Forward and Backward Functions

if two coders exhibit a certain *relative bias* (Wiebe et al., 1999) for a category C_1 (say, *Question*), and judgements on category C_2 (say, *Answer*) depend on the value chosen for C_1, they will presumably exhibit a correlated *relative bias* for *Answer*. The values of Kappa for the two categories will be correlated—as an accurate analysis of Kappa results in the early stages of development can also help in revising the coding instructions in a principled way (Wiebe et al., 1999), care should be taken that researchers focus on the independent categories. Only when the independent categories can be tagged reliably, does computing reliability for dependent categories make real sense (cf. the notion of *conditional reliability* (Krippendorff, 1980, ch. 12).

3. Using Kappa for evaluation

An interesting possibility to explore is whether Kappa can be used as a measure to evaluate systems in some fashion. The way coded data is often used is to train a system to infer the labels of interest. The final corpus on which the system is trained plays the role of the expert classification on that data (whether it is a single coder data, assembled from multiple coders, or a real "gold standard", cf. (Wiebe et al., 1999). Assuming that part of the coded data is set aside as a test set, Kappa could be used as an added measure apart from the percentage of test cases correctly classified to assess how well the system agrees with the expert classification. The system obviously is not just a clone of the coder or coders that hand tagged the training data, because learning algorithms introduce their own biases, and the data will no doubt contain noise (although possibly reduced by using techniques similar to what proposed by (Wiebe et al., 1999).) The system would then be evaluated in similar terms as psychiatry students who are being trained to diagnose schizophrenic patients.

4. Conclusions

In recent years, the Kappa coefficient of agreement has become the de facto standard to evaluate intercoder agreement in the discourse and dialogue processing community. Together with the adoption of this standard, researchers have adopted one specific scale to evaluate Kappa values, the one proposed in (Krippendorff, 1980), even if different scales, such as that by (Rietveld and van Hout, 1993), exist as well. In this position paper, I have highlighted some issues that should be taken into account when evaluating Kappa values. I have also speculated on whether Kappa could be used as a measure to evaluate a system's performance.

Future work clearly includes finding answers to the question I have raised in this paper.

5. References

Austin, John L., 1962. *How to Do Things With Words*. Oxford: Oxford University Press.

Berry, Charles C., 1992. The K statistic — to the editor. Letters, Journal of the American Medical Association, 268 (18).

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

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

DAMSL, 1997. DAMSL: Dialog act markup in several layers. Available under *Tools and resources*, at http://www.georgetown.edu/luperfoy/Discourse-Treebank/dri-home.html.

Di Eugenio, Barbara, Pamela W. Jordan, Johanna D. Moore, and Richmond H. Thomason, 1998. An empirical investigation of collaborative dialogues. In *ACL-COLING98, Proceedings of the Thirty-sixth Conference of the Association for Computational Linguistics*. Montreal, Canada.

Di Eugenio, Barbara, Pamela W. Jordan, Richmond H. Thomason, and Johanna D. Moore, 2000. The agreement process: An empirical investigation of human-human computer-mediated collaborative dialogues. *International Journal of Human Computer Studies*. To appear.

Goldman, Ronald L., 1992. The K statistic — to the editor (in reply). Letters, Journal of the American Medical Association, 268 (18).

Grove, William M., Nancy C. Andreasen, Patricia McDonald-Scott, Martin B. Keller, and Robert W. Shapiro, 1981. Reliability Studies of Psychiatric Diagnosis. Theory and Practice. *Archives of General Psychiatry*, 38:408–413.

Krippendorff, Klaus, 1980. *Content Analysis: an Introduction to its Methodology*. Beverly Hills: Sage Publications.

Passonneau, Rebecca J., 1997. Applying reliability metrics to co-reference annotation. Technical Report CUCS-017-097, Columbia University.

Rietveld, T. and R. van Hout, 1993. *Statistical Techniques for the Study of Language and Language Behaviour*. Mouton de Gruyter.

Searle, John R., 1965. What Is a Speech Act. In Max Black (ed.), *Philosophy in America*. Ithaca, New York: Cornell University Press, pages 615–628. Reprinted in *Pragmatics. A Reader*, Steven Davis editor, Oxford University Press, 1991.

Searle, John R., 1975. Indirect Speech Acts. In P. Cole and J.L. Morgan (eds.), *Syntax and Semantics 3. Speech Acts*. Academic Press. Reprinted in *Pragmatics. A Reader*, Steven Davis editor, Oxford University Press, 1991.

Siegel, Sidney and N. John Castellan, Jr., 1988. *Nonpara-*

metric statistics for the behavioral sciences. McGraw Hill.

Wiebe, Janyce M., Rebecca F. Bruce, and Thomas P. O'Hara, 1999. Development and use of a gold-standard data set for subjectivity classifications. In *ACL99, Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*. College Park, MD.