可選用中文或英文作答

A. (50%) Graph and Grammar

以下是 4 句不知名的語言片段，其單字英譯，與可能的句子翻譯 (不一定是唯一的翻譯)。

1. էրֆէրէրէ պկտ մե ֆրիէოენდ ოფ/ֆր ტოკყო́ֆ լոսդ

   dear/expensive gift my friend of/from Tokyo lost

   (An expensive gift to my friend from Tokyo was lost.)

2. Ի სეე ინ ըրպէն կարտէն լիդդლე გორლ0

   I See in urban garden little boy

   (I saw a little boy in the city garden.)

3. ტრონგ ჰუმან ეასილყ րաիսէ դհիս լոատդ

   Strong human easily raise this load

   (A strong man will easily lift this load.)

4. Սհէ Վիլլპუტ ტჰის არტիգլէ ინ ոըր պოხ

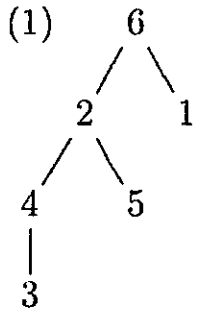
   She will_put this article in our box

   (She will put this article in our box.)

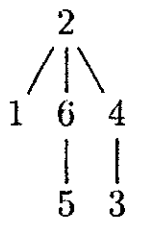
語言學家經過研究，發現這些句子都分別對應到以下的樹狀結構。請把這些對應找出，並提供說明。（注意，每一個句子對應到至少一個圖）。

見背面

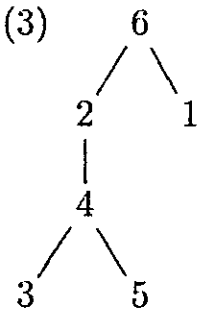
(1)



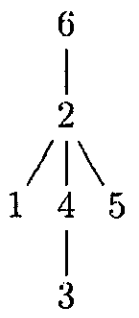
(2)



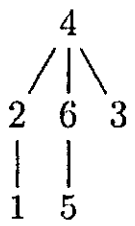
(3)



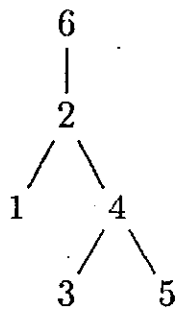
(4)



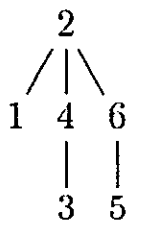
(5)



(6)



(7)



接次頁

B. (50%) inter-annotator agreement

An agreement coefficient calculates the amount that annotators agreed on label assignments beyond what is expected by chance. In a prototypical linguistic annotation task, annotators (a.k.s. coders) assign predefined *labels (or values)* to specific *units* (words, sentences, chunks, etc.) in the source. To measure the *reliability* of the annotation, a simple way of reporting agreement between annotators is called *raw agreement* measure which counts the number of items for which they provide identical labels, and report that number as a percentage of the total to be annotated.

Problem.1 (15%)

Why raw agreement does not imply that the annotation process is reliable? Provide examples to support your arguments.

In corpus and computational linguistics, a more accepted way is to measure the reliability of agreement by using a coefficient from the kappa/alpha family. For instance, **Krippendorff's alpha (α)** 's general form is:

見背面

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

where D_o is the observed disagreement among values assigned to units of analysis:

$$D_o = \frac{1}{n}\sum_c\sum_k o_{ck\,metric}\delta^2_{ck}$$

and D_e is the disagreement one would expect when the coding of units is attributable to chance rather than to the properties of these units:

$$D_e = \frac{1}{n(n-1)}\sum_c\sum_k n_c \cdot n_{k\,metric}\delta^2_{ck}$$

The arguments in the two disagreement measures, o_{ck}, n_c, n_k and n, refer to the frequencies of values in coincidence matrices, to be defined below.

(Krippendorff, 2011)

Take the following nominal data coded by two annotators for example, the computational steps involve:

1. Construct a 2-by12 rating table

Items judged:	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Ben:	*a*	*a*	*b*	*b*	*d*	*c*	*c*	*c*	*e*	*d*	*d*	*a*
Gerry:	*b*	*a*	*b*	*b*	*b*	*c*	*c*	*c*	*e*	*d*	*d*	*d*

2. Transform it into a coincidence matrix.

接次頁

Values:

	1	.	*k*	.	.	
1	o_{11}	.	o_{1k}	.	.	n_1
.	.		.			.
.	.		.			.
c	o_{u1}	.	o_{ck}	.	.	$n_c = \Sigma_k\, o_{ck}$
.	.		.			.
	n_1	.	n_k	.	.	$n = \Sigma_c \Sigma_k\, o_{ck}$

	a	*b*	*c*	*d*	*e*	
a	2	1	.	1	.	4
b	1	4	.	1	.	6
c	.	.	6	.	.	6
d	1	1	.	4	.	6
e	.	.	.	.	2	2
	4	6	6	6	2	24

Where $o_{ck} = \Sigma_u$ Number of *c*-*k* pairs in unit *u*

specifically:
$o_{ab} = 1$ *a*-*b* pair in unit 1
$o_{ba} = 1$ *b*-*a* pair in unit 1
$o_{aa} = 2$ *a*-*a* pairs in unit 2
$o_{bb} = 4 = 2$ *b*-*b* pairs in unit 3
$+ 2$ *b*-*b* pairs in unit 4
and so forth.
n_a=4 is the number of *a*s
n_b=6 is the number of *b*s
and so forth.
n =24 is the total number of values for two observers: $n = 2N$

3. Compute **alpha (α):**

$$_{\text{nominal}}\alpha = 1 - \frac{D_o}{D_e} = \frac{(n-1)\sum_c o_{cc} - \sum_c n_c(n_c-1)}{n(n-1) - \sum_c n_c(n_c-1)}$$

Problem.2 (35%)

We can use NLTK metrics package to calculate the inter-annotator agreement for this toy example. What'd be expected in the result? Calculate it manually based on the formulas.

```
from nltk.metrics.agreement import AnnotationTask
from nltk.metrics import binary_distance

annotation_triples = [('Ben', '1', '1'),
```

見背面

```
                    ('Gerry', '1', '2'),
                    ('Ben', '2', '1'),
                    ('Gerry', '2', '1'),
                    ('Ben', '3', '2'),
                    ('Gerry', '3', '2'),
                    ('Ben', '4', '2'),
                    ('Gerry', '4', '2'),
                    ('Ben', '5', '4'),
                    ('Gerry', '5', '2'),
                    ('Ben', '6', '3'),
                    ('Gerry', '6', '3'),
                    ('Ben', '7', '3'),
                    ('Gerry', '7', '3'),
                    ('Ben', '8', '3'),
                    ('Gerry', '8', '3'),
                    ('Ben', '9', '5'),
                    ('Gerry', '9', '5'),
                    ('Ben', '10', '4'),
                    ('Gerry', '10', '4'),
                    ('Ben', '11', '4'),
                    ('Gerry', '11', '4'),
                    ('Ben', '12', '1'),
                    ('Gerry', '12', '4')]

t = AnnotationTask(annotation_triples, distance = binary_distance)
result = t.alpha()
```

試題隨卷繳回