Probabilistic models for Pronunciation and Spelling

Probabilistic models for Pronunciation and Spelling

- In this Topic discusses the Problem of detecting and Correcting spelling errors.

a. First introduce the problems of detecting and Correcting spelling errors; also summarize typical human spelling error patterns

b. Introduce ways to solve the spelling problem : **Bayes Rule and the noisy channel model.**

Probabilistic models for Pronunciation and Spelling

- Dealing with spelling errors.
- Spelling error patterns.
- Detecting non word errors.
- Probabilistic model.
- Applying the Bayesian method to spelling
- Minimum edit distance

Dealing with spelling errors

- Classification of Spelling correction.
  1. **Non word error detection :** Detecting spelling errors that result in non-words.
  2. **Isolated-word error correction :** Correcting spelling errors that result in non words.(correcting graffe to giraffe, but looking only at the word in isolation.)
  3. **Context dependent error detection and correction :** using the context to help detect and correct real word errors.(dessert for desert or there for their)

Dealing with spelling errors

- Application area
  - Typed Text (Word Processor)
  - Optical character recognition – OCR (Optical scanner)
  - Online handwritten recognition

Spelling errors patterns

- The number and nature of spelling errors in human typed text differs from those caused by pattern recognition devices like OCR and handwriting recognizers.

  -**Number**

  - 1-3 % in human typed text.
  - Vary 0.2 -20% for OCR .

  -**Nature.**

Nature of Spelling errors

- Human typing errors

-Insertion : the as ther

-Deletion : the as th

- Substitution : the as thw

- Transposition : the as the

Nature of Spelling errors

- Other dimension of classification

- Typographic errors : keyboard related. Spell as spwll
- Cognitive errors : the writer doesn’t know how to spell. Separate as separate.

Nature of spelling errors

- OCR errors.

-Substitution

- Multi substitution
- Space deletion
- Insertion
- Failure

An example for OCR errors

- **Correct :** The quick brown fox jumps over the lazy dog.
- **Recognized :** ’lhe q~ick brown foxjurnps ovcr tb lazy dog.
- **Errors :** Substitution (e->c) and multisubstitutions (T-> ‘l, m -> rn, he -> b) are caused by visual similarity rather than keyboard distance; failure(u-> ~) are cases where OCR does not select any letter with sufficient accuracy.

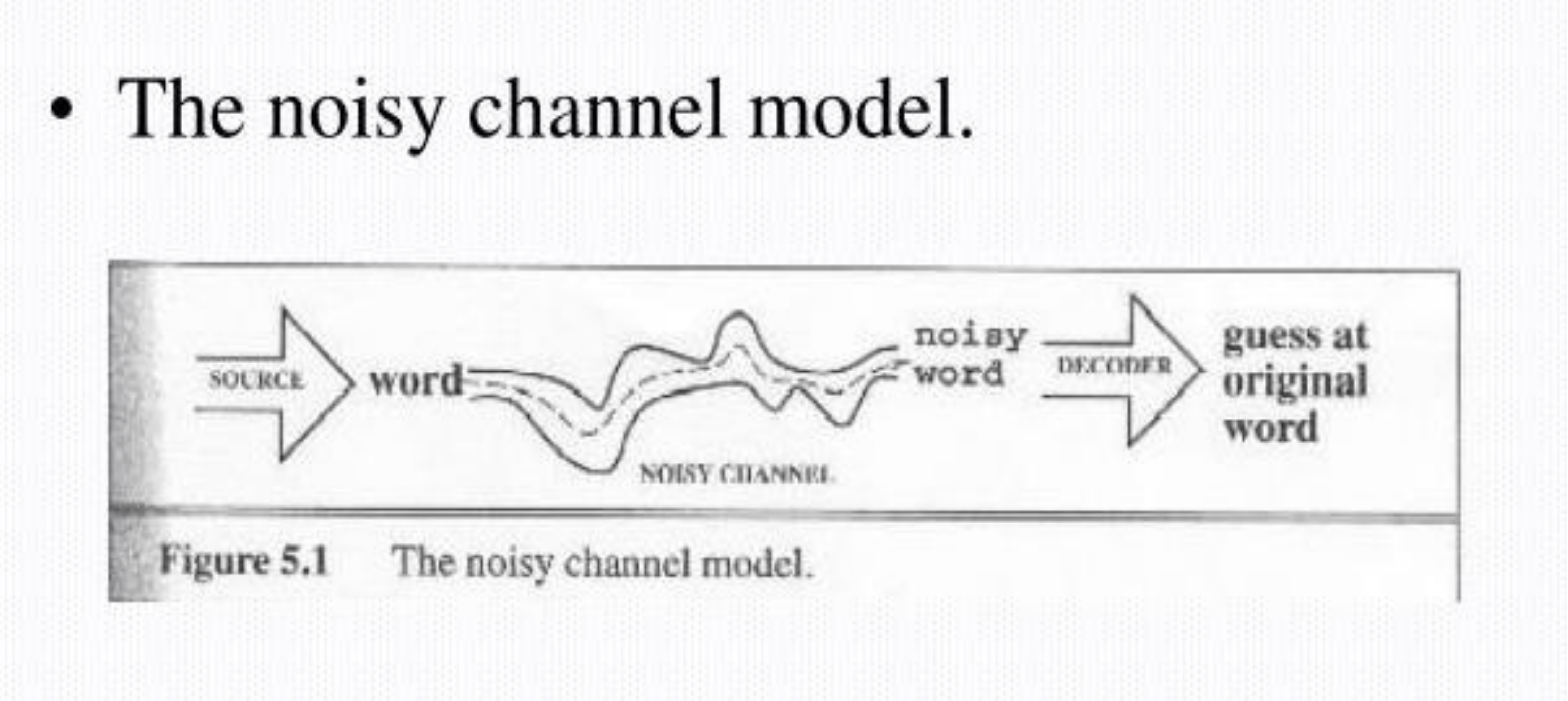
Detecting non-word errors

- Detecting non-word errors in text, whether typed by humans ro scanned, is commonly done by using dictionary.
- Small or big dictionary ?

  -small : Large dictionary contains rare words that resemble misspelling of other words : wont as won’t

  -Large : Emperical study found large dictionary are more helpful than harmful.
- Use model of morphology for to deal with inflection.

Probabilistic Model



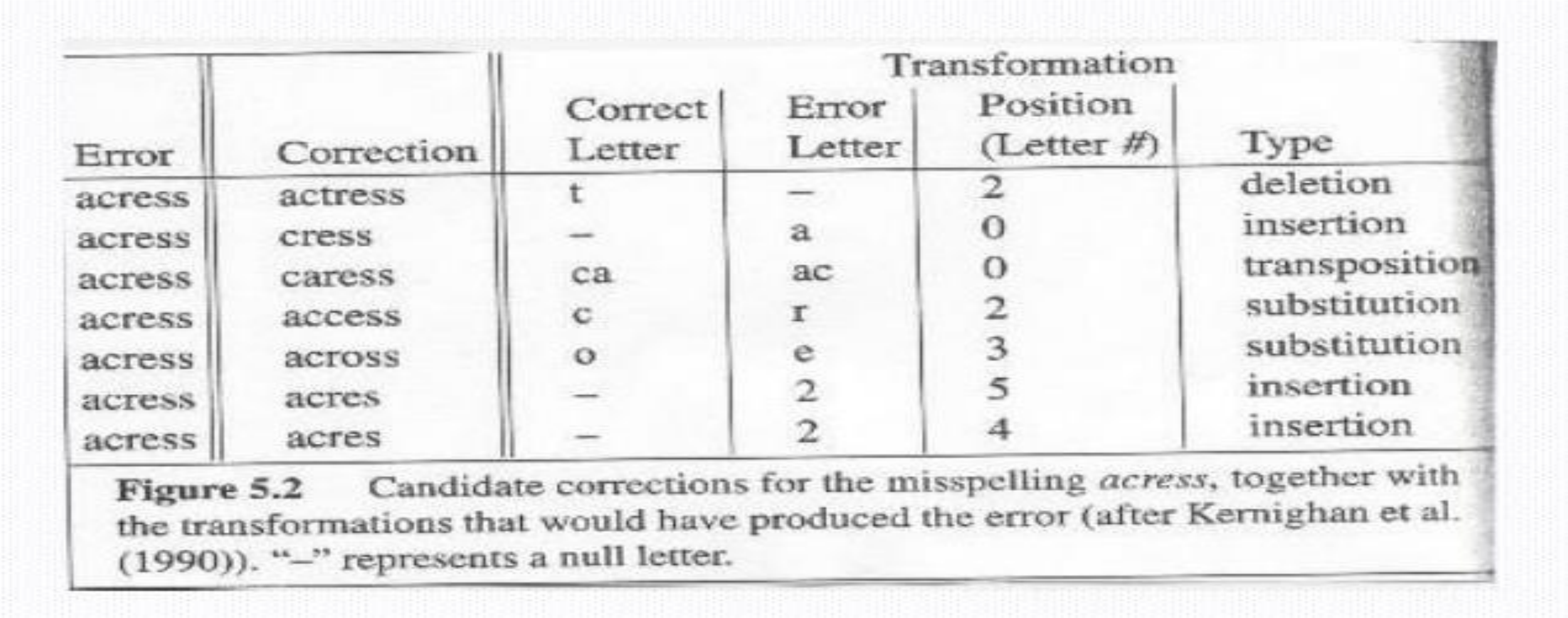
- The noisy channel model.

Figure 5.1 The noisy channel model.

Applying Bayesian Method

- Bayesian algorithm
  - -proposing candidate correlation
  - -Scoring the candidate
- Proposing the candidate
  - -Simplifying assumption: single spelling errors
  - -Example misspelling acress

Example



		Transformation			
Error	Correction	Correct Letter	Error Letter	Position (Letter #)	Type
acress	actress	t	–	2	deletion
acress	cress	–	a	0	insertion
acress	caress	ca	ac	0	transposition
acress	access	c	r	2	substitution
acress	across	o	e	3	substitution
acress	acres	–	2	5	insertion
acress	acres	–	2	4	insertion

Figure 5.2 Candidate corrections for the misspelling *acress*, together with the transformations that would have produced the error (after Kernighan et al. (1990)). "–" represents a null letter.

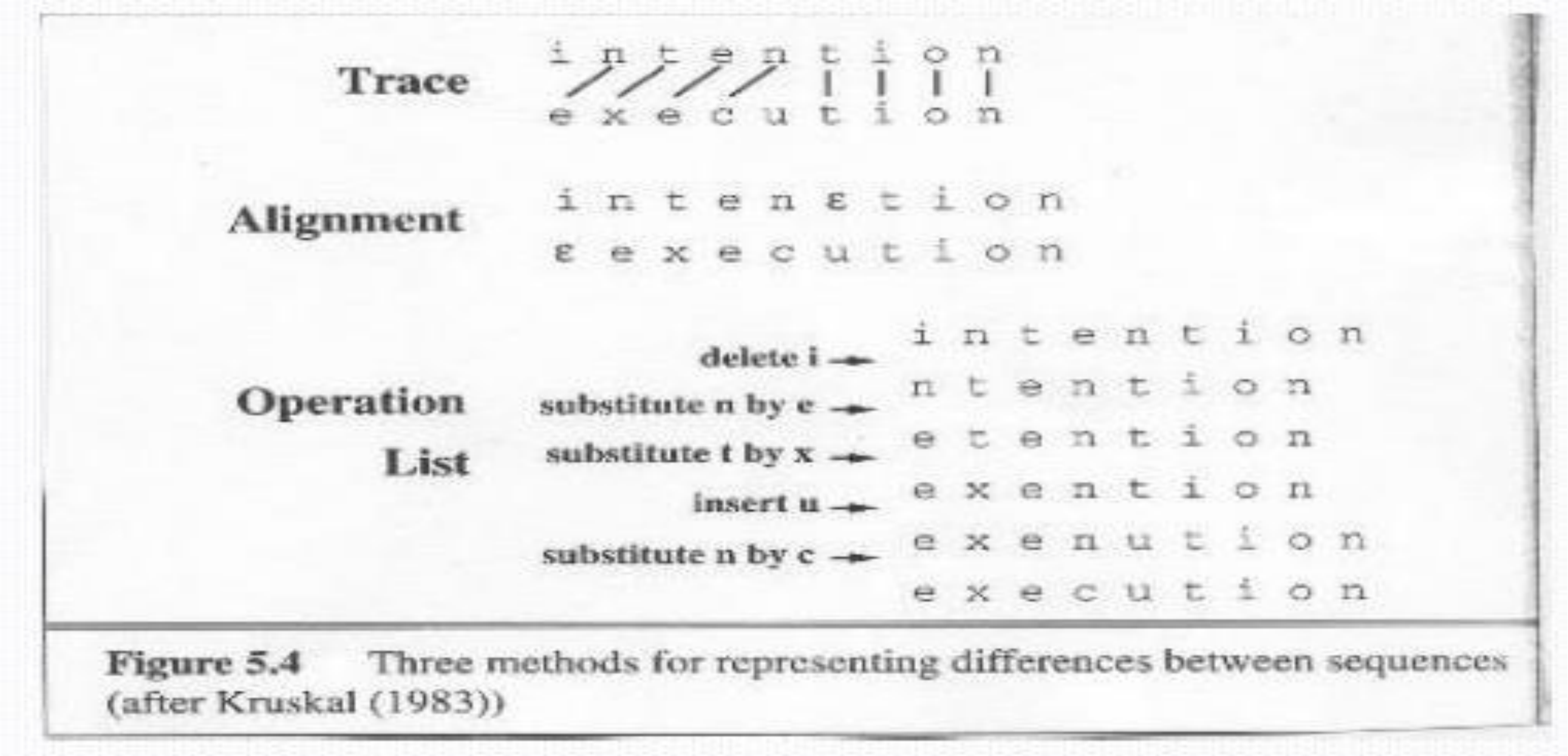
Minimum edit distance

- Previous section relied on the simplifying assumption- single spelling error.
- We need to more powerful algorithm to handle multiple errors.
- Minimum edit distance Algorithms

  -String distance , is some metric of how alike two strings are to each other.

  -The minimum edit distance between two string is the minimum number of editing operation.

Three method of Representing errors.



Trace

```
i n t e n t i o n
e x e c u t i o n
```

Alignment

```
i n t e n ε t i o n
ε e x e c u t i o n
```

Operation List

	i n t e n t i o n
delete i →	n t e n t i o n
substitute n by e →	e t e n t i o n
substitute t by x →	e x e n t i o n
insert u →	e x e n u t i o n
substitute n by c →	e x e c u t i o n

Figure 5.4 Three methods for representing differences between sequences (after Kruskal (1983))