2.1 Parts of speech

本节介绍词性。某些词在句子中具有相同的性质，在它们之间互换不改变句子结构，这些词具有同样的词性。可以简单分为开类型（open, lexical）和闭类型（close, functional），前者随时间不断扩充，如名词、动词，后者在语言中是恒定的，如限定词（determiner, 包括冠词 article，如 the）、介词（preposition, in, on）。

对于陌生的单词，我们可以根据上下文（context）推测其词性，但仍可能出现含糊不清（ambiguous）的情况。对于计算机而言这将更困难，当它遇到陌生的单词或需要明确某个多义单词的含义时：

- They have to use some prior knowledge. 必须利用先验知识
- They reason probabilistically. 推测概率
- They use context. 结合上下文
- The can be wrong. 可能犯错，而且错误可能会有小到大地蔓延

英语中的词性列举如下：

1. 名词 Nouns，可数名词有单复数形式；

2. 代词 Pronouns，如 she, ourselves, mine，形式有主格（nominative，I）、宾格（accusative，me）、所有格（形容词性 possessive，如 my，名词性 2nd possessive，如 mine），以及反身代词（reflexive，herself），anaphoric（each other）；

3. 限定词 Determiner， 包括冠词 Articles（the, a）和指示词 Demonstratives（this, that）；

4. 形容词 Adjectives，用于描述性质，可以作定语 attributive（如 small house）或表语 predicative（如 The house is small）。此外，形容词有比较级（comparative）和最高级形式（superlative），可以派生（derivative）而来（如 smaller），或外加词（periphrastic）而来（如 more difficult）；

5. 名词 Verbs，描述动作 actions、活动 activities 和状态 states，如 throw, walk, have，针对不同时态语态情境，动词具有更多形式和用法；

6. 副词 Adverbs，如 happily, here, never；

7. 介词 Prepositions，如 of, through, in；

8. 小品词 Particles，是构成短语动词 phrasal verbs 的一部分，如 the plane took off, take it off。为了具有完整的含义短语动词是不可分割的，例如 take off 不是某种特别的 take 方式（is

not some special way if taking），而是一个完整的独立的动词（a completely different verb）。再如 He ran up a bill（账单，这句话表示他欠债了），He ran up a hill，简单的区分方式是 Up a bill, he ran 不合语法，因此这里 run up 不可分割，up 是小品词；而 Up a hill, he ran 合语法，因此这里 up 是介词；

9. 并列连词 Coordinating conjunctions，and, or, but，连接地位平等的句子；

10. 从属连词 Subordinating conjunctions，if, because, that, although，连接地位不等即具有从属关系的句子；

11. 感叹词 Interjections，如 Ouch !

2.2 Morphology and the Lexicon

语言形态学使得我们可以按照一定规律极大地丰富词汇。根据相应的规律和习惯我们可以推测派生词的含义，但有时会有含糊的情况，例如 undoable 可以理解为 unable to be done 或 able to be undone。此外，不同语言中形态学规律不同，有些语言可以通过重复某些字母组合来表现复数，英语口语有时将某单词插入另一单词的重音左侧形成组合词。

词素（morpheme）是单词中形态学意义上的单元（individual units of morphological meaning），一般包括词干（stem）和词缀（affix）。词缀一般又包括前缀（prefix）、后缀（suffix）、词尾（ending，如-ing），对应英语等语言中的拼接形态（concatenative morphology）。其他语言中还有在词中插入的情况（templatic morphology），如 Semitic 语言中，lmd (learn), lamad (he studied), limed (he taught)，在不同位置插入不同的元音（vowel）可以生成不同含义的变形。

单词的变形（inflectional morphology）也是形态学的重要部分。单词常常因下列因素而变形：

时态 tense，现在 present、过去 pass、将来 future；

数量 number，单数 singular、复数 plural；

人称 person，第一人称 I、第二人称 you、第三人称 he；

语气 mood，陈述语气（indicative）、虚拟语气（subjunctive）、条件语气（conditional）等；

体 aspect，如进行 progressive、完成 perfective。

英语动词的变形较少，单个单词最多 5 中变形（am, are, was, were, been），但其他语言可能更多。如法语 40 多种，俄语 6 种。

形态分析（Morphological Analysis）是将单词转换为它的形态表达的过程。例如：

sleeps = sleep（原形 infinitive）+ V（动词词性）+ 3P（第三人称）+ SG（单数）

done = do（原形）+ V（动词词性）+ PP（过去分词 past participle）

随后举例说明土耳其语（Turkish）中形态与元音和谐（vowel harmony）的例子。背景知识是，Turkish 中元音可以分为 back vowels（形成于口的前部，be formed in the back of the mouth）和 front vowels，为了满足元音和谐一个原则是一个单词中只能有其中一类的元音（within the same word, you can only have back vowels or front vowels）。Turkish 中后缀 da 和 de 用于表示位置（in, on, or at），其中 da 用于含 back vowels 的单词，de 用于 front。例如 odada（in the room）、kapıda（at the door），evde（at home）。

Radev 举例说明了英语单词或句子转换为土耳其词汇或日语词汇的例子，这一过程遵循目标语言的规则和习惯。

语义学（semantics）研究单词或句子的含义，可以分为词汇语义学（lexical semantics）和成分语义学（compositional semantics）。

词汇语义学关注的一些问题如下：同义词（synonym）、反义词（antonym），多义词（polysemous word），词的搭配（collocation, a sequence of words that appear together more frequently than you would expect if you just combine the probabilities of the individual words）如 stock market，习语（idiom, specific collocation that have a meaning that is not literally inferable from the components）如 to kick the bucket（直译踢桶，实际表示 to die）。

成分语义学研究如何基于句子成分理解句子含义。此外，后续课程还将介绍语用学。

最后，Radev 列举了几个与本课程关系不大的几个语言研究领域。

2.3 Introduction of Text similarity

文本相似性是 NLP 中的重要组成部分。以下是该技术涉及到的几个应用例子：

1. 能够识别同一含义的不同表述，如“the plane leaves at 12pm”和“the flight departs at noon”；

2. 能够识别相关联的概念，比如 cat 和 kitten 含义相近，而 fruit dessert（甜点）包括 peach tart 和 apple cobbler，这对于信息检索而言很重要；

3. 能够识别发音或拼写相近但含义不同的词汇，如 Dulles 和 Dallas 是完全不同的机场名字。

研究和应用中常常用数字来量化相似性，人工评判的结果可以用于衡量自动计算的相似性是否合理，但研究表明不同人评判的结果总体具有较大的方差，因此其可靠性也需考虑。

最后介绍了几种常见的相似性类型，详细参见 ppt。相关名词：synonymy 同义，homophony 同音异意，paraphrase 意译、改述，lingual 语言的。

2.4 Morphological Similarity: Stemming

形态相似性的度量经常通过词干提取（stemming, to take a word and to convert it into a base form）的方式进行。这一过程的初衷在于词干相同的词汇经常具有相似的含义。

一种经典的方法是 Porter's stemming method，提出于 1980 年的论文《An algorithm for suffix stripping（除去）》。这是一种基于规则的算法，而所有规则都是人工生成的，不借助机器学习或训练。该算法只应用于英语，不处理前缀，且不保证结果总是完全正确的。

为了执行该算法，首先定义单词的度量（the measure of a word），它近似表示单词的音节数（the number of syllables）。任何单词都可以表示为[C](VC){k}[V]的形式，其中 C 代表辅音，V 代表元音，且连续出现的辅音或元音序列只计一次（即结果中不会出现 CC 或 VV 的情况）。于是，k 即单词的度量值。

Porter 算法是一个迭代的过程，每次迭代都包含 4 步，而每步都按照一定顺序执行一系列规则的检验。具体而言，原始单词首先进入第一步，并与第一步中的第一条规则所需的模式条件进行匹配，若单词中有一部分与其匹配成功且单词未被匹配的部分的度量大于 0（例如，第一步中有规则 EED -> EE，会使得 refereed -> referee，但不会令 bleed -> blee，因为 m('bl')=0），则执行该规则的转换，否则继续与下一条规则进行匹配，若整个第一步的规则都没能匹配则进入下一步。

因此每轮迭代可能出现两种情况，成功匹配某一步的某一规则且满足度量约束，则将转换后的新单词重新从第一步开始处理。否则，4 个过程中的规则全不能匹配，则算法结束，当前的单词即为词干。

2.5 Spelling Similarity:Edit Distance

拼写相似性的常用度量是编辑距离。该相似性一个常用应用是存在少量拼写错误时识别正确的拼写形式，另一个重要应用是当某序列存在多义性的时候给出最或然的含义。

编辑距离的计算经常涉及的 4 种操作：

1. 插入 insertion 字母；
2. 删除 deletion 字母；
3. 替换 substitution 字母；
4. 交换相邻字母 a swap of adjacent letters，一些情况下可能不考虑这种操作。

Levenshtein Method 用于计算只涉及前三种操作的编辑距离，它是一种动态规划算法，一般将每种操作的代价均设为 1。对于待计算的两个字符串 s1 和 s2，该算法指出对于两段前缀 p1（s1 前缀）和 p2（s2 前缀），其编辑距离等于两前缀前一状态的编辑距离加上对应的编辑操作所引入的新的距离。结合算法描述和图 1 可以直观地理解该算法。

对于该算法（结合图 1 的生成过程更容易理解），有两点需要说明：

1. t(i,j)=0 出现后，对任意 k>j 有 t(i,k)=1 恒成立，也就是说如果两个单词的某个字符已经匹配成功了，那么这个字符不应该再参与后续的匹配成功。对应图 1，黄色框的右侧是严格单调递增的，不存在匹配成功的情况；

2. 由 1 可见，匹配成功的位置是重要的，因此需要在算法过程中记录下来，在恢复匹配流程时，匹配路径不应当经过成功位置的左侧和上方（起始位置除外），其含义是该字符在匹配成功前未被使用。

- Definitions
  - $s_1(i)$ – i^{th} character in string s_1
  - $s_2(j)$ – j^{th} character in string s_2
  - $D(i,j)$ – edit distance between a prefix of s_1 of length i and a prefix of s_2 of length j
  - $t(i,j)$ – cost of aligning the i^{th} character in string s_1 with the j^{th} character in string s_2
- Recursive dependencies

$$D(i,0)=i$$
$$D(0,j)=j$$
$$D(i,j)=\min[D(i-1,j)+1,\ D(1,j-1)+1,\ D(i-1,j-1)+t(i,j)]$$

- Simple edit distance:

$$t(i,j)=0 \ \textit{iff}\ s_1(i)=s_2(j)$$
$$t(i,j)=1, \textit{otherwise}$$

算法描述

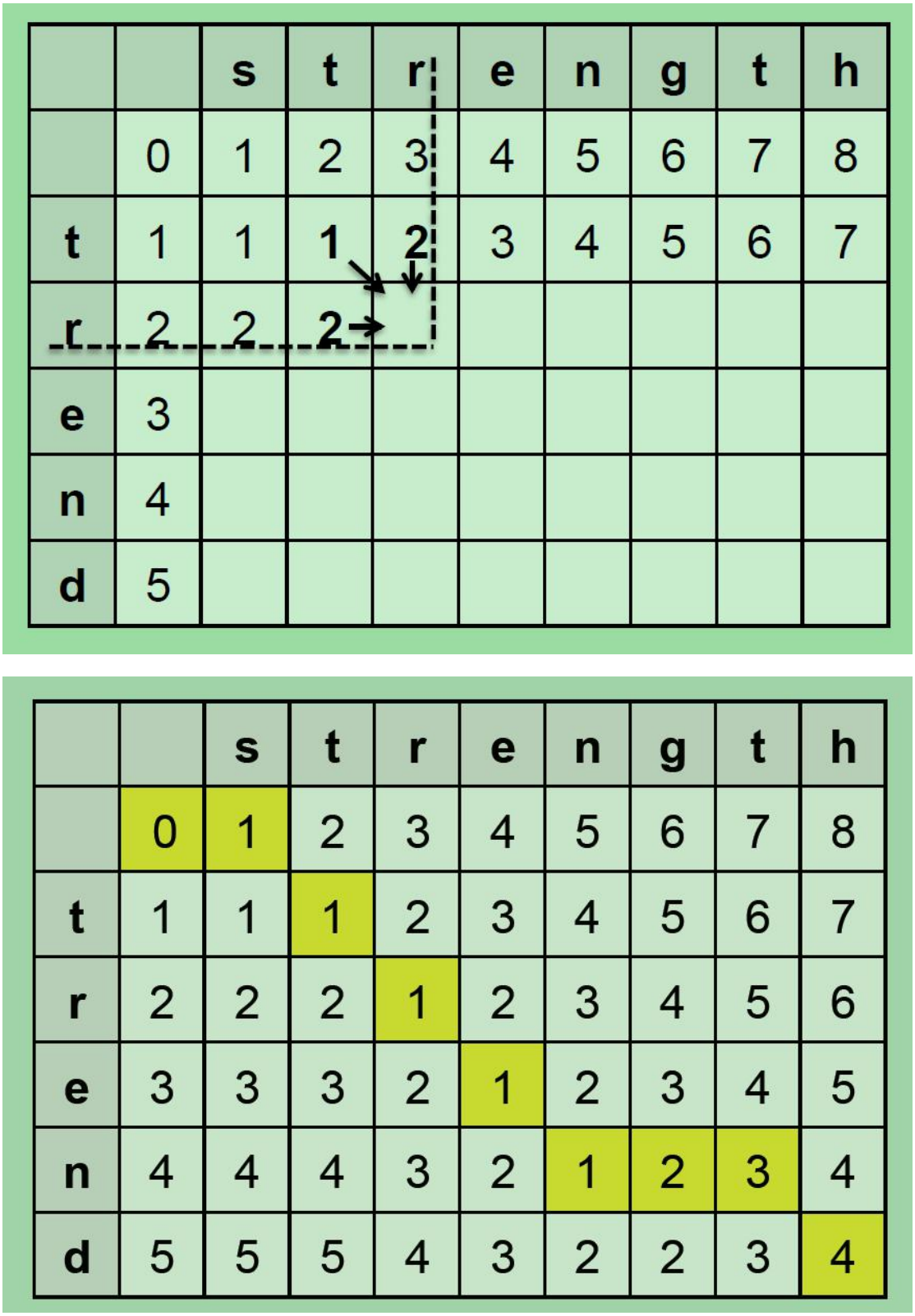

图 1 算法某状态的示意图

除了基本的 Levenshtein 编辑距离外，还可根据具体应用设计更合理的距离。例如考虑到相邻字母的交换（这里简称 swap）是一种常见的拼写错误，Damerau 距离将 swap 操作的

代价设为 1，在 Levenshtein 中 swap 的代价相当于 2。其他的两个例子如下：

1. 在手写体识别中，d 和 cl 之间容易混淆，因此可以有 Dist (“sit clown”,“sit down”) = 1；

2. 键盘输入时，鉴于 w 与 q 相距很近，而与 l 相距很远，可以令 Dist (“qeather”,” weather”) = 1，而 Dist(“leather”,”weather”) = 2。

此外，编辑距离不仅可以用于计算文本之间的相似性，也可应用于其他类似对象。例如碱基序列、氨基酸序列等。

2.6 Introduction of NACLO

简要介绍了北美计算语言学奥林匹克（North American Computational Linguistics Olympiad）。展示了一些有代表性的问题，给出了一些有趣问题的链接。该课程过程中的一些问题就来自于该竞赛。

2.7 Preprocessing

为了便于后续 NLP 过程，原始的数据一般需要预处理。常见的预处理过程涉及到：

1. 处理字符编码问题 Dealing with text encoding (e.g., Unicode)；

2. 去除多余数据 Removing non-text (e.g., image, ads, javascript)；

3. 句子划分 Sentence segmentation；

4. 标准化 Normalization，某些单词可能具有不同的变种（multiple variants），例如 labeled（英式）/labelled（美式），extra-terrestrial/extraterrestrial/extra terrestrial，我们一般希望将它们转变为相同的形式（same form）以避免混乱；

5. 词干提取 Stemming，computer/computation->comput；

6. 形态分析 Morphological analysis，car/cars；

7. 大写字母处理 Capitalization 以及实体名称识别 Named entity extraction，Now/NOW, led/LED。

预处理过程中，句子划分（也称句子边界识别 Sentence Boundary Recognition）会遇到一些问题。基本的出发点是标点符号标识句子边界（punctuation symbols indicate the end of a sentence），常见的标识结尾的符号如句号 period。但是一些符号不止出现在句子结尾，以

句号为例，还可能出现在句子中，如 Dr. Willow、A.L.S。为了考虑这些因素，一种处理方式是利用决策树 decision trees，按照一定的规则顺序识别是否为边界。例如英语中，若句号后不紧跟空格，则不认为此处是步子结尾。

对中文、日语等语言而言，词分割（Word Segmentation）更为困难。

此外，其他的一些困难还存在于名词实体、数字（包括电话号码、车牌、门牌）、不同格式的日期、URL 等。