1 n-gram による特徴抽出

n-gram は、n 個の記号の並びからなるデータ構造です。
「This is a pen.」を解析対象データだと解釈し、n を 3 として、3-gram を作成すると、

Thi
his
is_
s_i
_is
is_
s_a
...
が、3-gram になります。(空白を_で表しています。)
n-gram を用いると、基となる文章の特徴を学習することが可能です。
例えば、文章全体から n-gram を作成し、個々の n-gram の出現頻度を数えます。すると、n-gram の頻度分布が、ある文章の特徴を表します。
「はじめての機械学習」小高知宏著　60 頁

2 n-gram を作成するプログラム

ngram.c

```
/*
  ngram.c
  ngram を作成するプログラム
  入力行は標準入力から与え、出力は標準出力に出力します。
  使い方
  ./ngram (N の値) < (入力ファイル名) > (出力ファイル名)
  入力ファイルには、テキストファイルを指定します。
  出力ファイルには、ngram を出力します。
*/

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define N 32

void setlastch(int n, char chr, char lastch[]);

void printngram(int n, char lastch[]);
```

```
int main(int argc, char *argv[])
{
  int  i;
  int  n;
  char lastch[N] = {' '};
  char chr;

  if(argc != 2){
    fprintf(stderr, "使い方  $./ngram (N の値) "
                    " < (入力ファイル名) > (出力ファイル名)\n");
    exit(1);
  }

  if(((n = atoi (argv [1])) < 1) || (n >= N)){
    fprintf(stderr, "N の値が不適切です。\n");
    exit(1);
  }

  while((chr = getchar()) != EOF){
    if(chr != '\n'){
      setlastch(n, chr, lastch);
      printngram(n, lastch);
    }
  }

  return 0;
}

void setlastch(int n, char chr, char lastch[])
{
  int i;

  for(i = n - 2; i >= 0; --i){
    lastch[i + 1] = lastch[i];
  }
  lastch[0] = chr;
}

void printngram(int n, char lastch[])
{
  int i;

  for(i = n - 1; i >= 0; --i){
    printf("%c", lastch[i]);
  }
  printf("\n");
}
```

3 「ngram」の操作方法

「ngram」の操作方法の例

```
./ngram 5 < alice.txt > alice5gram.txt
```

上記の「./」は、Linux でのプログラムを実行する時の例です。
ウィンドウズでは、ngram 5 < alice.txt > alice5gram.txt「Enter」
と入力してください。

4 n-gram の頻度解析

rank.c

```
/*
  rank.c
  ngram の頻度分布を作成します
  入力行は標準入力から与え、出力は標準出力に出力します。
  使い方
  ./rank <（入力ファイル名）>（出力ファイル名）
  入力ファイルには、ngram のファイルを指定します。
  出力ファイルには、頻度分布を出力します。
*/

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define N 32
#define ITEMNO 65535*3
#define NUMF 4

int uniq(int n, char ngram[ITEMNO][N],char result[ITEMNO][N + NUMF]);

int rescmp(const char *a, const char *b);

char ngram[ITEMNO][N];
char result[ITEMNO][N + NUMF];

int main(int argc, char *argv[])
{
  int  i,j;
  int  no;

  i = 0;

  while(fgets(ngram[i] , N, stdin) != NULL){
```

```
    if(i >= ITEMNO){
      fprintf(stderr, "ngram が最大数に達しました。\n");
      break;
    }
    ++i;
  }

  qsort(ngram, i, N, (int (*)(const void *, const void *))strcmp);

  no = uniq(i, ngram, result);

  qsort(result, no, N + NUMF, (int (*)(const void *, const void *))rescmp);

  for(j = 0; j < i; ++j){
    printf("%s", result[j]);
  }

  return 0;
}

int uniq(int n, char ngram[ITEMNO][N],char result[ITEMNO][N + NUMF])
{
  int i,j;
  int c;
  char lastgram[N];

  strncpy(lastgram, ngram[0], N);
  j = 0;
  c = 1;

  for(i = 1; i < n; ++i){
    if(strcmp(lastgram, ngram[i]) == 0){
      ++c;
    }else{
      sprintf(result[j], "%d\t%s", c, lastgram);
      strncpy(lastgram, ngram[i], N);
      c = 1;
      ++j;
    }
  }

  sprintf(result[j],  "%d\t%s", c, lastgram);
  strncpy(lastgram, ngram[i], N);
  ++j;
  return j;
}

int rescmp(const char *a, const char *b)
{
```

```
  int numa, numb;

  numa = atoi(a); numb = atoi(b);

  if(numa > numb) return -1;
  else if(numa < numb) return 1;
  return 0;
}
```

5 「rank」の操作方法

「rank」の操作方法の例

```
./rank < alice5gram.txt > alice5rank.txt
```

上記の「./」は、Linux でのプログラムを実行する時の例です。
ウィンドウズでは、rank < alice5gram.txt > alice5rank.txt 「Enter」
と入力してください。

6 「alice5rank」の実行結果の例

「alice5rank」の実行結果の例

```
1476  the
688  and
459 d the
447  said
431 , and
423  she
403 said
398 Alice
329  Alic
324 ’ sai
```