機械翻訳

東京大学　情報基盤センター
情報理工学系研究科、情報学府　兼担）
中川裕志

昔の機械翻訳

- 入力文：私はりんごを食べた。
-
  - →形態素解析→構文解析
  - noun verb noun → subj predicate object
    - →意味解析
    - (action=食べる, agent=私, target=りんご, time=past)
      - 英語の語彙に変換（つまり意味表現のレベルないしはそれに近い深さで変換　←対訳辞書利用
        - (action=eat, agent=I, target=an apple, time=past)
      - 構文および形態素の生成（語順の変換）して翻訳出力を得る。←対訳辞書利用
  - noun=I, verb（past)=ate, noun=an apple
- 出力文：I ate an apple.

昔の機械翻訳

- 意味のレベルで精密に日英が同一であることが前提だった。
- また、形態素解析、構文解析、意味解析が正確に動作すると想定している。
- しかし、なかなかそうとも言い切れない
  - 意味レベルでの概念が一致しない例
    - 湯　→ hot water、
    - もったいない→? 、
    - checkという習慣が日本にない！

対訳辞書

- 日本語→意味
  - りんご → APPLE
- 意味→英語
  - ALLPE→ if bear noun or singular: apple
    if plural: apples

◆単数の場合には an apple,複数なら apples を選ぶのは、構文および形態素のレベル

少し前の機械翻訳：example based translation

- 翻訳対の例文が類似検索可能な形でデータベース化
  - 例：私はみかんを食べた。←→ I ate an orange.

- 入力文：私はりんごを食べた。
  - 翻訳対データベースから類似した日本語例文を検索
    - 私はみかんを食べた。
      - 違っている部分みかんをりんごに置き換え
    - さらに日英辞書でりんごをan appleに置き換え

- 結果出力：I ate an apple.
  - 当然ながら、冠詞の選択などは文法規則によって行う。つまり、相当程度に従来の構文規則や、形態素解析技術と共同することになる。

少し前の機械翻訳：example based translation

- 類似検索の部分が重要。ここで構文解析を使うことも可能だが、だんだん古典的な機械翻訳に近づく。
- 翻訳対を集めれれば集めるほどが翻訳の質があがる。
  - この収集作業は機械的にできる。

統計的機械翻訳
Statistic Machine Translation (SMT)

- 言語的知識を全く使わずに対訳を得る。アンチ言語学理論
- 2言語並行コーパスが蓄積
- 文どうしの対応付けされた aligned corpus
- これを使って単語や句どうしの対応付け、すなわち対訳を自動的に抽出
  - 文同士の対応はあるが、単語列同士の対応は不明
  - 探索空間が膨大
- IBMの Peter Brown, S. Della Pietra, V. Della Pietra, Robert Mercerらの1993年のCLの論文“The Mathematics of Statistical Machine Translation:Parameter Estimation”を中心に解説

Bayesの定理

- Canadian Hansard : French-English Bilingual corpus
- フランス語の単語列:*f* に対して妥当な英語の単語列 : *e* を求める
- Given French string: *f*, find $e^{\wedge}=\arg\max_e \Pr(e|f)$
  - 種々の*f*に対応しそうな*e*はやたらと多い！！
- then

$$\mathbf{Pr}(\boldsymbol{e}|\boldsymbol{f}) = \frac{\Pr(\boldsymbol{e})\Pr(\boldsymbol{f}|\boldsymbol{e})}{\Pr(\boldsymbol{f})}$$

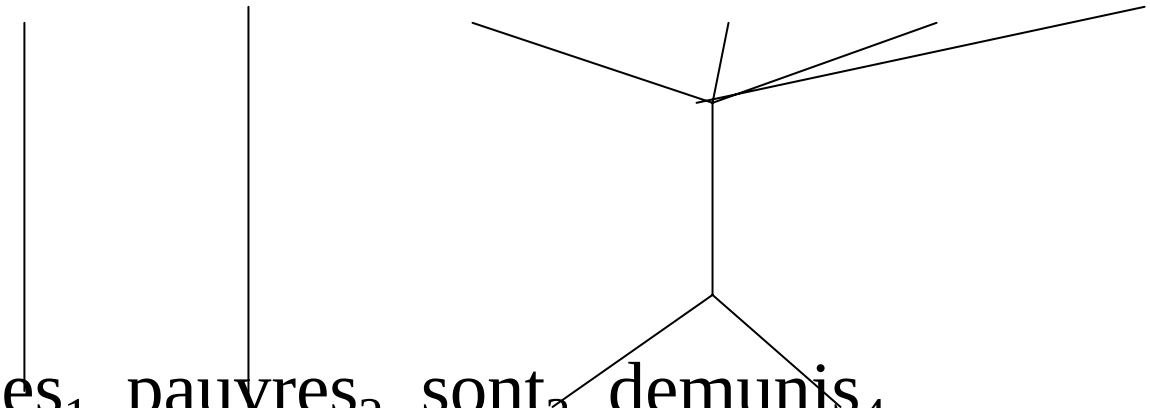
$$\Rightarrow \quad \hat{\boldsymbol{e}} = \underset{\boldsymbol{e}}{\operatorname{argmax}} \Pr(\boldsymbol{e})\Pr(\boldsymbol{f}|\boldsymbol{e})$$

なぜPr($e|f$)ではなく、Pr($f|e$)×Pr(e)か？

- 種々のfに対応しそうなeはやたらと多い！！
  - 対訳コーパスの対訳文はやはり少数
- 無尽蔵に多くあるフランス語の単語列 f に対して、対応すべき正しい英語を求めるのが目的
- Pr($e|f$)直接では、正しい英単語列eに高い確率が割り当てられることが保証されない。
- 正しい英文という要因を直接考慮するためにPr(e)を別個の情報源から得て利用する。

Alignment：対応

- The$_1$ poor$_2$ don't$_3$ have$_4$ any$_5$ money$_6$



- Les$_1$ pauvres$_2$ sont$_3$ demunis$_4$

(Les pauvres sont demunis |

The(1) poor(2) don't(3,4) have(3,4) any(3,4) money(3,4))

=A(e,f)=**a**

→ e,f**はここでは文**

記法

- Alignmentも考慮したPr(f,a|e)

$$\Pr(\mathrm{f}\,|\,\mathrm{e}) = \sum_{\mathrm{a}} \Pr(\mathrm{f},\mathrm{a}\,|\,\mathrm{e})$$

$\mathrm{e} = e_1^l \equiv e_1 e_2 .... e_l$ *where l English words*

$\mathrm{f} = f_1^m = f_1 f_2 .... f_m$ *where m french words*

$\mathrm{a} = a_1^m = a_1 a_2 .....a_{m,}$ *where* $a_j = i$

f,e は単語列 a は*alignment* f_i, e_{a_j}は単語

$$\Pr(\mathrm{f},\mathrm{a}\,|\,\mathrm{e}) = \Pr(m\,|\,\mathrm{e}) \prod_{1}^{\mathrm{m}} \Pr(a_j\,|\,a_1^{j-1}, f_1^{j-1}, m, \mathrm{e}) \Pr(f_j\,|\,a_1^{j}, f_1^{j-1}, m, \mathrm{e})$$

- 以後はPr(f,a,|e)を評価する方法

IBM Model 1

- このモデルでは、英、仏文の単語の出現順序には相関がないとしている。ー(1)
- また対訳は個々の単語にだけ依存するー(2)

$$\varepsilon \equiv \Pr(m \mid \mathrm{e})$$

$$\Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, m, \mathrm{e}) = (l+1)^{-1} \qquad -(1)$$

$$\Pr(f_j \mid a_1^{j-1}, f_1^{j-1}, m, \mathrm{e}) = t(f_j \mid e_{a_j}) \qquad -(2)$$

$$\Pr(\mathrm{f}, \mathrm{a} \mid \mathrm{e}) = \frac{\varepsilon}{(l+1)^m} \prod_1^{\mathrm{m}} t(f_j \mid e_{a_j}) \qquad -(3)$$

f, e　は単語列　　a　は*alignment*　　f_i, e_{a_j}は単語

Model 1

- このモデルでは、Alignment a_j は0から m の任意の値をとるから$1/(l+1)$。ラグランジュ未定乗数法によってPr(f|e)を最大化する。

$$\Pr(\mathrm{f}\,|\,\mathrm{e}) = \frac{\varepsilon}{(l+1)^m} \sum_{a_{1=0}}^{l} \cdots \sum_{a_{m=0}}^{l} \prod_{1}^{\mathrm{m}} t(f_j \,|\, e_{a_j}) \qquad -(4)$$

$$\text{constraint}: \sum_{f} t(f \,|\, e\,) = 1 \qquad f, e\text{は単語}f_j, e_{a_j}\text{のいずれかを表す。}$$

$$h(t,\lambda) \equiv \frac{\varepsilon}{(l+1)^m} \sum_{a_{1=0}}^{l} \cdots \sum_{a_{m=0}}^{l} \prod_{j=1}^{\mathrm{m}} t(f_j \,|\, e_{a_j}) - \sum_{e} \lambda_e \Big(\sum_{f} t(f \,|\, e\,) - 1\Big) \qquad -(5)$$

$$0 = \frac{\partial h}{\partial t(f\,|\,e)} = \frac{\varepsilon}{(l+1)^m} \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} \sum_{j=1}^{m} \delta(f, f_j)\delta(e, e_{a_j}) t(f\,|\,e)^{-1} \prod_{k=1}^{\mathrm{m}} t(f_k \,|\, e_{a_k}) - \lambda_e \qquad -(6)$$

$$t(f \mid e) = \lambda_e^{-1} \frac{\varepsilon}{(l+1)^m} \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} \sum_{j=1}^{m} \delta(f, f_j)\delta(e, e_{a_j}) \prod_{k=1}^{m} t(f_k \mid e_{a_k})$$

$$= \lambda_e^{-1} \sum_{a} \Pr(\mathrm{f}, \mathrm{a} \mid \mathrm{e}) \sum_{j=1}^{m} \delta(f, f_j)\delta(e, e_{a_j}) \qquad -(7)$$

$$c(f \mid e; \mathrm{f}, \mathrm{e}) = \sum_{a} \Pr(\mathrm{a} \mid \mathrm{f}, \mathrm{e}) \sum_{j=1}^{m} \delta(f, f_j)\delta(e, e_{a_j}) \qquad -(8)$$

- $c(\ldots)$ とは翻訳(f|e)において、英単語e が フランス語単語f に翻訳される回数。2番目のΣはあるalignment a においてf,eの対訳された回数で総和。

$\Pr(\mathrm{a} \mid \mathrm{f}, \mathrm{e}) = \Pr(\mathrm{f}, \mathrm{a} \mid \mathrm{e}) / \Pr(\mathrm{f} \mid \mathrm{e})$　　f, a, e　は単語列

$$\rightarrow c(f \mid e; \mathrm{f}, \mathrm{e}) = \frac{\sum_{a} \Pr(\mathrm{f}, \mathrm{a} \mid \mathrm{e}) \sum_{j=1}^{m} \delta(f, f_j)\delta(e, e_{a_j})}{\Pr(\mathrm{f} \mid \mathrm{e})} - (9)$$

$$\rightarrow t(f \mid e) = \lambda_e^{-1} \Pr(\mathrm{f} \mid \mathrm{e})\, c(f \mid e; \mathrm{f}, \mathrm{e}) \qquad -(10)$$

- (9)式の $\sum_{j=1}^{m}\delta(f,f_j)\sum_{i=0}^{l}\delta(e,e_i)$ の部分は*f*と*e*の対訳回数になることが分かる。(alignment aがないのでこの式)下図参照。

- **f**= f1, f2(=f), f3, ..,f7(=f), …. fm
- e1(=e) * *
- e2
- **e**= :
- e8(=e) * *
- :
- el

- 教師データとしてS個の翻訳 $(f^{(s)}|e^{(s)})$ s=1,…,Sがコーパスから知られているので、S個の翻訳の総和を用いて式(10)を書き換えた以下の式を使うので覚えておいてください。ただし、(10)の $\lambda_e Pr(f|e)$ を λ_eと置き換えた。

$$t(f|e)=\lambda_e^{-1}\sum_{s=1}^{S}c(f|e;f^{(s)}e^{(s)})$$

$t(f|e)$を求めるまではもう一工夫

- $t(f_j|e_{a_i})$ は、単項式だから

$$\sum_{a_{1=0}}^{l}\cdots\sum_{a_{m=0}}^{l}\prod_{j=1}^{\mathrm{m}}t(f_j|e_{a_j})=\prod_{j=1}^{m}\sum_{i=0}^{l}t(f_j|e_i) \qquad -(11)$$

- 例 $t_{10}t_{20}+t_{10}t_{21}+t_{11}t_{20}+t_{11}t_{21}=(t_{10}+t_{11})(t_{20}+t_{21})$
- これによると

$$\Pr(\mathrm{f}|\mathrm{e})=\frac{\varepsilon}{(l+1)^m}\prod_{j=1}^{m}\sum_{i=0}^{l}t(f_j|e_i)-(12)$$

◆そこで、またラグランジュ未定乗数法で

$$\text{constraint}: \sum_{f} t(f \mid e) = 1$$

$$h(t,\lambda) \equiv \frac{\varepsilon}{(l+1)^m} \prod_{j=1}^{m} \sum_{i=0}^{l} \sum_{c} t(f_j \mid e_i) - \sum_{e} \lambda_e \left(\sum_{f} t(f \mid e) - 1 \right) \quad -(13)$$

$$0 = \frac{\partial h}{\partial t(f \mid e)} = \frac{\varepsilon}{(l+1)^m \sum_{i=0}^{l} \sum_{c} t(f_j \mid e_i)} \prod_{j=1}^{m} \sum_{i=0}^{l} \sum_{c} t(f_j \mid e_i) \sum_{j=1}^{m} \delta(f, f_j) \sum_{i=1}^{l} \delta(e, e_i) - \lambda_e$$

$$-(14)$$

$$\rightarrow \lambda_e^{-1} \Pr(\mathrm{f} \mid \mathrm{e}) = \frac{\sum_{i=0}^{l} t(f_j \mid e_i)}{\sum_{j=1}^{m} \delta(f, f_j) \sum_{i=0}^{l} \delta(e, e_i)} \quad -(15)$$

$$by(10)(15) \rightarrow \qquad c(f \mid e; \mathrm{f}, \mathrm{e}) = \frac{t(f \mid e)}{t(f \mid e_0) + \ldots + t(f \mid e_l)} \sum_{j=1}^{m} \delta(f, f_j) \sum_{i=0}^{l} \delta(e, e_i) - (16)$$

いよいよEMで$t(f|e)$を推定－1

1. $t(f|e)$の初期値を適当に決める
2. 各$(f^{(s)},e^{(s)})$, 1=<s =<Sに対して、

$$c(f|e;f^{(s)},e^{(s)})=\frac{t(f|e)}{t(f|e_0)+\ldots+t(f|e_l)}\sum_{j=1}^{m}\delta(f,f_j)\sum_{i=0}^{l}\delta(e,e_i)$$

を計算する。

$\sum_{j=1}^{m}\delta(f,f_j)\sum_{i=0}^{l}\delta(e,e_i)$ の値は f,e が$f^{(s)},e^{(s)}$の要素のときだけ0でない。

いよいよEMで$t(f|e)$を推定－2

3. $$t(f|e) = \lambda_e^{-1} \sum_{s=1}^{S} c(f|e;\mathrm{f}^{(s)}\mathrm{e}^{(s)})$$ を $\sum_f$ すると

左辺が1になるので、

$$\lambda_e = \sum_f \sum_{s=1}^{S} c(f|e;\mathrm{f}^{(s)},\mathrm{e}^{(s)})$$

この λ_e の値を用いて$t(f|e)$の新たな値を推定する。

$$t(f|e) = \frac{\sum_{s=1}^{S} c(f|e;\mathrm{f}^{(s)},\mathrm{e}^{(s)})}{\sum_f \sum_{s=1}^{S} c(f|e;\mathrm{f}^{(s)},\mathrm{e}^{(s)})}$$

4. $t(f|e)$が収束するまで2,3を繰り返す。

Model 2

- Alignmentが位置に依存する。つまり、

$$a(a_j \mid j, m, l) \equiv \Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, m, l)$$

が　j　a_j　m　l　に依存

$$\sum_{i=0}^{l} a(i \mid j, m, l) = 1$$

then　$$\Pr(\mathrm{f} \mid \mathrm{e}) = \varepsilon \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} \prod_{j=1}^{m} t(f_j \mid e_{a_j}) a(a_j \mid j, m, l)$$

ラグランジュ

$$h(t, a, \lambda, \mu)$$

$$= \Pr(\mathrm{f} \mid \mathrm{e}) - \sum_{\mathrm{e}} \lambda_{\mathrm{e}} \left(\sum_{f} t(f \mid e) - 1 \right) - \sum_{j} \mu_{jml} \left(\sum_{i} a(i \mid j, m, l) - 1 \right)$$

ラグランジュ未定乗数法でhを微分し計算すると

$$t(f \mid e) = \lambda_e^{-1} \sum_a \Pr(f, a \mid e) \sum_{j=1}^{m} \delta(f, f_j)\delta(e, e_j)$$

$$c(f \mid e; \mathrm{f}, \mathrm{e}) = \sum_a \Pr(\mathrm{a} \mid \mathrm{f}, \mathrm{e}) \sum_{j=1}^{m} \delta(f, f_j)\delta(e, e_{a_j})$$

$$t(f \mid e) = \lambda_e^{-1} \Pr(f \mid e)\, c(f \mid e; \mathrm{f}, \mathrm{e})$$

$$t(f \mid e) = \lambda_e^{-1} \Pr(f \mid e) \sum_{s=1}^{S} c(f \mid e; \mathrm{f}^{(s)}\, \mathrm{e}^{(s)})$$

ここまでは同じだが、　　さらに意味的に考えて

$$c(i \mid j, m, l; \mathrm{f}, \mathrm{e}) = \sum_a \Pr(\mathrm{a} \mid \mathrm{e}, \mathrm{f})\delta(i, a_j)$$

$$a(i \mid j, m, l) = \mu_{jml}^{-1} \sum_{s=1}^{S} c(i \mid j, m, l; \mathrm{f}^{(s)}, \mathrm{e}^{(s)})$$

注：$\Pr(a \mid f, e) \Pr(f \mid e) = \Pr(a, f \mid e)$

$\mu_{jml}\ \Pr(f \mid e) \rightarrow \mu_{jml}$

Model 1と同じように計算し

- *Model 1* では$(l+1)^{-1}$ だった*a(i|j,m,l)*を*Model 2* では変数と見ているので、

$$\mathbf{Pr}(\mathrm{f}\,|\,\mathrm{e}) = \varepsilon \prod_{j=1}^{m} \sum_{i=0}^{l} t(f_j\,|\,e_i)\,a(i\,|\,j,m,l)$$

$$c(f\,|\,e;\mathrm{f},\mathrm{e}) = \sum_{j=1}^{m} \sum_{i=0}^{l} \frac{t(f\,|\,e)\,a(i\,|\,j,m,l)\,\delta(f,f_j)\,\delta(e,e_i)}{t(f\,|\,e_0)\,a(0\,|\,j,m,l) + \ldots + t(f\,|\,e_l)\,a(l\,|\,j,m,l)}$$

$$c(i\,|\,j,m,l;\mathrm{f},\mathrm{e}) = \frac{t(f_j\,|\,e_i)\,a(i\,|\,j,m,l)}{t(f_j\,|\,e_0)\,a(0\,|\,j,m,l) + \ldots + t(f_j\,|\,e_l)\,a(l\,|\,j,m,l)}$$

- 後は同じくEMアルゴリズムで*t(f|e)*を求める
- 初期値にはModel 1の結果を用いる。

Model 3

- １単語がｎ単語に翻訳　not => ne … pas
- ｎ＝０（翻訳されない）の場合も考慮。　冠詞は日本語にはない。
- 対応する単語の出現場所がねじれる
  - 日英での語順の差
- こういった現象に対応するモデル

- 繁殖確率 $n(\phi|e)$: 英語単語eがϕ個のフランス語単語に接続される確率
- 翻訳確率t(f|e):英語単語eがフランス語単語fに翻訳される確率
- 歪確率d(j|i,m,l):英文長さl,フランス文長さm,英文の単語位置iがフランス文の単語位置jに接続される確率
- 空の英単語の繁殖数＝ ϕ_0

- 空でない英単語から生成されたフランス語単語の後に空から生成されたϕ_0個の単語が確率p_1で挿入されるとすると、

$$\Pr(\ \phi_0 \mid \phi_1^{\ l}, e\ ) = \begin{pmatrix} \phi_1 + .. + \phi_l \\ \phi_0 \end{pmatrix} p_0^{\ \phi_1 + .. + \phi_l - \phi_0}\ p_1^{\ \phi_0}$$

$$p_0 + p_1 = 1$$

$$\sum_{i=0}^{m} \phi_i = m$$

以上の準備の下

$$\Pr(\mathrm{f}\mid \mathrm{e}) = \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} \Pr(\mathrm{f}, \mathrm{a}\mid \mathrm{e})$$

$$= \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} \binom{m-\phi_0}{\phi_0} p_0^{m-2\phi_0} p_1^{\phi_0} \prod_{i=1}^{l} \phi_i! n(\phi_i \mid e_i) \times$$

$$\prod_{j=1}^{m} t(f_j \mid e_{a_j}) d(j \mid a_j, m, l) \qquad -(32)$$

$\phi_i!$ はe_iから生成されたϕi個の単語列における順番を入れ替えた単語列の数え上げ

- (32)式を用いて、n,t,d,$p_{0,1}$に関する各々の総和＝1の確率による条件をつけてラグランジュ未定乗数法で Pr(e|f) を最大化すればよい。
- しかし、model1,2と異なり和積の交換ができないので、直接計算する。
- 組み合わせの数が多いので、ビタビ法で近似計算する。

二言語コーパスからの対訳抽出 -- Aligned corpus の場合--

- Parallel Corpus（平行、対訳コーパス）
  - Aligned Corpus: 種々の研究あり。要はどのようにして2つの言語のコーパスにおける文、単語、複合語、句を 対応付ける(align) するかに集中。
  - ９０年代前半にきれいな2言語対訳コーパスを対象にした多数の研究があり。
  - ９０年代後半に、Noisy Parallel Corpus　への展開が試みられた (Fung94,Fung98)

対訳コーパスからの対応文のペアを求める

- Gale and Church 1993
- 2言語の文書　S,Tから対応付け(Alignment) Aを求める。
- SとTの対応する文のペアを bead という。
  - 例 *B*=(言語　language),

    B=(les eaux mineral, mineral water)
- $Alignment = argmax_A\ P(A|S,T) = argmax_A\ P(A,S,T)$
- B_kは 文書先頭からk番目のbead

$$P(A,S,T) \approx \prod_{k=1}^{K} P(B_k)$$

対訳コーパスからの対応文のペアを求める

- 例 B=(言語　language),

  B=(les eaux mineral, mineral water)

$$\boldsymbol{Align} = \mathbf{argmax}_{A}\, \boldsymbol{P}(\boldsymbol{A},\boldsymbol{S},\boldsymbol{T}) \approx \mathbf{argmax}_{A} \prod_{\boldsymbol{k}=1}^{\boldsymbol{K}} \boldsymbol{P}(\boldsymbol{B_k})$$

- これらから分かるように、よい対応付けは、単語やcollocationレベルでの2言語での対応付けから得られるbead: B_kの確からしさP(B_k)が大きいものを集めて作る。
- 具体的なアルゴリズムはDPによる

対訳コーパスからの対応文のペアを求める

- このようにして、文内部の対応がうまく付く文同士が高い対応付け確率を持つとする。
- 文書同士で文同士が高い対応付けを持つペアを連ねるように選ぶと対応文ペアが求まる。
  - 文書を構成する各文同士の対応付けもまた、文の対応付け確率を用いたDPによる。
- Beadを作る単語同士の対訳の他に、文の長さが類似しているほど、よく対応する文のペアと考える方法も加味できる。

対訳コーパスからの対訳文の概要

- 動的計画法による探索　Gale&Chruch93, Nissen et al 98
- A*探索　Wang & Waibel 97
- 確実な対訳と見られるアンカー点を決めつつの反復法 Kay & Roscheisen 93

Aligned corpus からの単語やcollocation 対訳抽出の概要

- 対訳判定の尺度:
  - 各言語の対訳文毎に対訳かどうか判定したい表現 w1,w2（＝各言語から抽出された表現）が出現する文としない文を数える。
  - w1,w2 が対訳文の出現する確率による。ただし、この確率＝0ならEMアルゴリズムで推定。Kupiec93
  - w1,w2の相互情報量　Haruno93, Smadja96
  - w1,w2のDice係数　Smadja96
  - ヒューリスティック
  - Likelihood ratio　Melamed97

語の2言語コーパスでの共起とその有意さの尺度

- あるテキストの単位（文、段落など）へのW1（言語A),w2（言語B)のcontingency matrix （頻度） a+b+c+d=N

	W2出現	W2非出現
W1出現	a	b
W1非出現	c	d

- 相互情報量 $MI(w1,w2)=\log\frac{P(w1,w2)}{P(w1)P(w2)}=\log\frac{aN}{(a+b)(a+c)}$

- Dice係数 $Dice(w1,w2)=\frac{2a}{(a+b)+(a+c)}$

Champollion (Smdja et al 96)

- Translating collocations： based on sentence aligned bilingual corpus
- 1つの言語においてXtractで　collocation を抽出
- 相手側の言語の対応文である統計的フィルタをパスするような分布で出現している collocation を対訳とする。
- フィルタとしては、相互情報量、Dice係数を比較した結果、

Champollion

- Dice係数は、X=0,Y=0(双方の言語の文にcollocationの要素がない場合)に影響されないので、精度が高いので、これを使う。

$$Dice(X,Y) = \frac{2\,freq(X=1,Y=1)}{freq(X=1)+freq(Y=1)}$$

- 大雑把には collocation (の要素たち)のDice係数が閾値以上のもののみを残す。
- なお、極端に出現頻度の低い要素語は捨てる。

Champollion

- Canadian Hansards (50MB order)
  - 3000, 5000 collocations extracted by Xtract
  - 中頻度の300 collocations で評価
  - Xtractのerror rate = 11%
  - Incorrect translations = 24%
  - Correct translations = 65%
  - Champollion’s precsion = 73%

Likelihood ratio: Melamed 97

- Melamed 97 の方法は、対訳はone-to-oneという仮定をおき、
  - ２言語の単語u,vが対訳であるかどうかの尤度比を計算し、
  - 最大の尤度比のものを対訳とする。
  - ただし、u,vの個別の出現例から
  - λ +=P(対訳|共起かつ対訳)、λ -=P(対訳|共起かつ非対訳)、をコーパスで評価しておき
  - λ + λ - を∏u,vP(k(u,v)|n(u,v), λ ±)を最大化するようにして求める
- 次の尤度比を用いる。
  L(u,v)=B(k(u,v)|n(u,v), λ +)/B(k(u,v)|n(u,v), λ -)
  - λ ± 機能語、内容語で分類する
  - recall=90% -- precision=87%

二言語コーパスからの対訳抽出 -- non aligned corpus の場合--

- Non-aligned Corpus の場合
  - Alignさせながら対訳を探す方法 Fung95ACL
  - Alignment を使わず別の情報源による方法
    - 複合語の構成 Fung95WVLC
    - 文脈に出現する語の分布
      Rapp95,Tanaka96 ,Fung98,
    - 特定の分野のコーパスであること

二言語コーパスからの対訳抽出 -- non aligned corpus の場合--

- 基本的方法は、まず双方の言語で用語あるいは collocation を抽出し、次に何らかの方法で対応をつけようとする。

Noisy Parallel Corpus の場合

- Fung94 ACL では English-French parallel corpus (Hansards) から次のような方法で対訳を求めた。
  1. 同じ内容をもつ英仏テキスト対をK等分する
  2. 英単語Wenと仏単語WfrのK個のセグメントでの現れ方（＜1，0，1，0，0＞などと表現）の相関を相互情報量MIで測る
  3. MIの高いものが対訳とみなす。
  4. 低頻度単語でMIが高くでる傾向の是正策（t-scoreによる検定）も行う。
- K分割というところが味噌。Fung95につながる。

高頻度の確実な対訳語対を探してAlignment させながら低頻度語の対訳も探す方法 ——Noisy Parallel Corpus の場合——

- Fung95 ACL95
- Alignment と 対訳抽出を同時に行う。
- Step1. 両言語のコーパスで頻度の高い語についての対応付けをする。うまくいけば、この対応付けが同時に対訳になる。
  - 各語の相対出現位置ベクトルを求める
  - このベクトルをDP風にマッチさせ対応する対を選ぶ。なお、ベクトルの平均と分散のによるユークリッド距離の離れている対は除去
  - この対応付けをアンカー点と呼ぶ
  - 高頻度の対応付けだけを残すとかなりきれいな alignment になっている

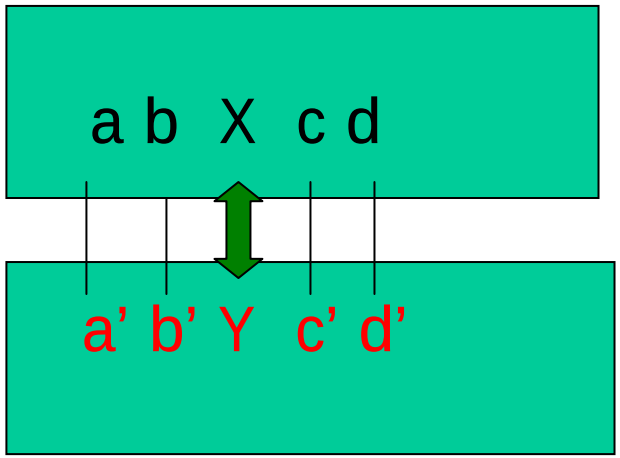
Fung 95 ACL の続き

- Step2: 低頻度の語の対応付け
  - アンカー点で区切ったセグメントをs1,s2,…とする（K等分ではないところが進歩）
  - 低頻度の語の相対位置ベクトルをセグメントの順番(i of si)の列に変換する。これによって、ラフな alignment でも誤差がセグメント単位に切り上げられる。
  - その上で相互情報量の小さい対を対訳として選ぶ。

Fung 95 ACL の続き

- 結果
  - 6000語ほどの小さな英中コーパス
  - 高頻度の対訳　128語　80%弱の精度
  - 低頻度の対訳　533語　70%程度の精度
  - 平均して　73%の精度
- 必ずしも実用レベルの実験ではない

非平行な同一分野の2言語コーパスからの訳語抽出

- Non-parallel comparable corpora
- Similarity of context is cue.
- 

language A
- language B
- Calculation of contexts similarity is heavy

Context Heterogeneity (Fung95WVLC)

- 扱う分野が同じなら parallel でなくてよい。
- 単語 trigram により文脈を作る。つまり
- 単語L 単語0 単語R
  - ここで、単語0 の context heterogeneity は
  - 単語Lの異なり数＝a, 単語Rの異なり数＝b
  - 単語0 の出現頻度=c のとき
  - Left-heterogeneity=a/c, rigth-heterogeneity=b/c
  - このアイデアは語基の連接数（中川）に似ている
- 二つの言語の単語w1,w2の context heterogeneity x1,y1 (for w1), x2,y2 (for w2)の距離は

$$\varepsilon=\sqrt{(x1-x2)^2+(y1-y2)^2}$$

Context heterogeneity の評価

- ε は統計量なのでとにかくコーパスが大きくないと信頼性の高いデータが得られない。
  - 大規模なデータ量は、non-aligned corpus の場合、必然かも。
- Context heterogeneity のような局所的な文脈だけで対訳を絞りこめるのか疑問
  - 評価結果が論文に書いてない
- 既存の辞書を使っていない。ゼロから対訳を得ようというのはかなり無理では。

より本格的に文脈を利用する方法　Rapp

- Rapp95,99 では、共起する語の分布の近さで訳語の曖昧性を解消しようとする。
- 小規模な英独対訳辞書は利用する。
- ドイツ語コーパスである単語wdの前後２単語づつでの共起語を使う。
- 上で述べた位置関係において二つの単語 wa,wb　の共起を頻度や確率、MIでなく次にしめす対数尤度比（これが最良と報告）によってあらわす。

より本格的に文脈を利用する方法　Rapp

対数尤度比

- あるテキストの単位（文、段落など）へのwa,wb（言語B)のcontingency matrix （頻度） a+b+c+d=N,
- wa,wbが共起= a, waだけ= b ,wbだけ= c,両方ともない=d

$$-2\log\lambda = a\log\frac{aN}{(a+b)(a+c)} + b\log\frac{bN}{(a+b)(b+d)}$$

$$+ c\log\frac{cN}{(a+c)(c+d)} + d\log\frac{dN}{(b+d)(c+d)}$$

- この対数尤度比をベクトルの要素にして、waに共起するw(ger)とその対訳w(eng)のベクトルの距離を次式で測る

$$s(w(ger),w(eng)) = \left|(\lambda_{-2}\lambda_{-1}\lambda_{+1}\lambda_{+2})_{w(ger)} - (\lambda_{-2}\lambda_{-1}\lambda_{+1}\lambda_{+2})_{w(eng)}\right|$$

より本格的に文脈を利用する方法　Rapp

- 前式でw(ger)の対訳w(eng)は既存の辞書で求める。
- 距離s(w(ger),w(eng))をwaに近接する語基全てについて加算し、これをSとする。
- Sの最小のものがwaの訳語であるとする。
- ドイツ語の語基から英語への対訳を100語のサンプルで評価した結果、
- 第1順位の対訳の精度は72%、
- 上位10位以内に正しい訳語がある場合も加えると89%の精度

より本格的に文脈を利用する方法　Fung

- Fung98 : Rappと同様のアイデア
- 既存の中英対訳辞書を利用して中国語未知語Wの英訳語を求める。
- Wの出現した文と同じ文における共起する既存の辞書の単語　Wdをひとつの次元とみなすベクトル空間法
- Wと同じ文におけるWdの出現数をtf
- Wと同じ文をdocument と見なした場合のidf (Wと同じ文に出現する総単語数で正規化

より本格的に文脈を利用する方法　Fung

- tf･idf を直接Wdの重みにせずに、Wdの英訳語の既存の辞書での重要度順位kで割る。
- cosine, Dice係数などで類似度の高い英単語をWdの対訳とする。
- 第１位の訳語の精度は３０％
- 第２０位までの訳語だと７６％の正解をカバー

より本格的に文脈を利用する方法　Tanaka　K

- Tanaka96　類似性はMI、辞書はEDICT
  - 言語A　　T　　言語B
  - 単語u　→　単語k
  - A　↕　　↕　T^AT vs B
  - 単語v　→　単語l
  - Tij=p(言語Bの単語j | 言語Aの単語i)
  - |T^AT－B|　を最急下降法で最小化し、求めたTが対訳マトリクス。
- ３７８語の対訳実験で、
  曖昧さ解消成功率 ＝８２％。一方、ふるい落とした対訳の８５％は誤訳語

Context Heterogeneity (Fung95WVLC)

- 扱う分野が同じなら parallel でなくてよい。
- 単語 trigram により文脈を作る。つまり
- 単語L 単語0 単語R
  - ここで、単語0 の context heterogeneity は
  - 単語Lの異なり数＝a, 単語Rの異なり数＝b
  - 単語0 の出現頻度=c のとき
  - Left-heterogeneity=a/c, rigth-heterogeneity=b/c
  - このアイデアは語基の連接数（中川）に似ている
- 二つの言語の単語w1,w2の context heterogeneity x1,y1 (for w1), x2,y2 (for w2)の距離は

$$\varepsilon=\sqrt{(x1-x2)^2+(y1-y2)^2}$$

Context heterogeneity の評価

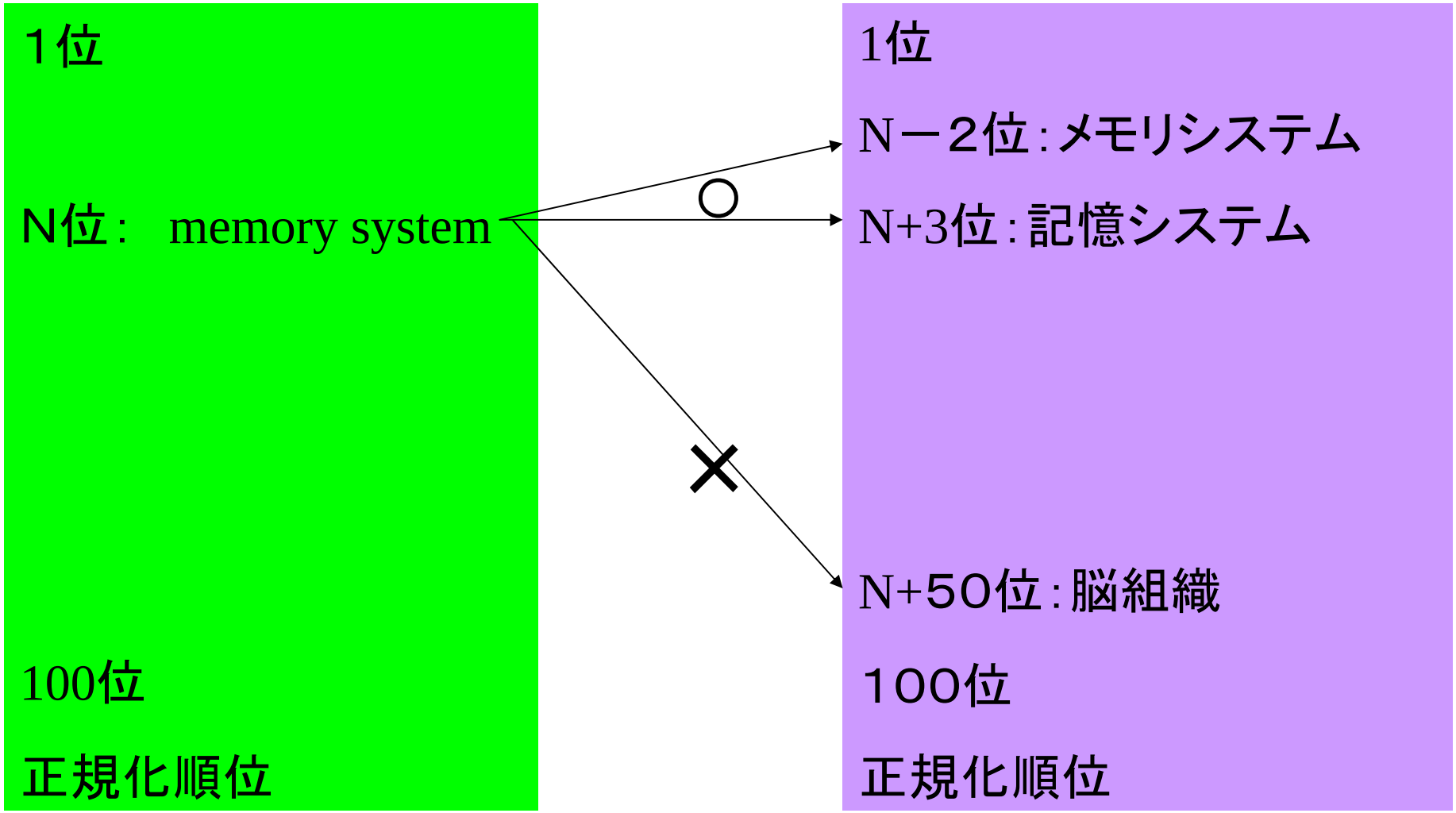
- ε は統計量なのでとにかくコーパスが大きくないと信頼性の高いデータが得られない。
  - 大規模なデータ量は、non-aligned corpus の場合、必然かも。
- Context heterogeneity のような局所的な文脈だけで対訳を絞りこめるのか疑問
  - 評価結果が論文に書いてない
- 既存の辞書を使っていない。ゼロから対訳を得ようというのはかなり無理では。

既存の対訳辞書と語基の連接数との併用
nakagawa2000 LREC WTRC, 2001 NLPRS

- 各言語のコーパスから用語候補を抽出
- 言語毎に、抽出された語基を連接数の大きさの順に並べる
- 日本語の語基wjの英語の対訳候補we1,we2,..を既存の対訳辞書（EDICT)から求める。当然、曖昧
- wj の対訳として、wj の連接数の順位に近い連接数の順位を持つwei (i=1,2,..) を選ぶ

抽出、順位つけされた
用語集合（英語）

抽出、順位つけされた
用語集合（日本語）



Distance

- distance(Xe, Xj) = | rank(Xe)-rank(Xj) |

- If distance(Xe, Xj) is small, Xe is the translation of Xj.
- distance(Xe1,Xj)<distance(Xe2,Xj)<…

  then Xe1 is most likely translation of Xj

Example of distance

memory system	メモリシステム	0.051493
	記憶システム	0.956459
	メモリ方式	1.234347
	記憶方式	3.809609
	脳組織	63.498688

日英，英日の情報通信関連専門用語の対訳で60%から80%の精度

より本格的に文脈を利用する方法　まとめ

- Rapp、Fung, Tanaka の方法は
  1. タネになる小さな対訳辞書によって
  2. 対訳を求めたい語基の文脈（に現れる既知語）をベクトル表現し
  3. 相手側テキストで、このベクトルによく一致するものを探し、そこの未知語を訳語にする
- この方法は、行き着くところまで行っている（local minimum）ので、新規なアイデアが必要？？