2G4-OS19a-6

大規模高次Nグラムを用いて動作文生成を行う運動認識システム

Motion Recognition System Generating Motion Sentence
using Large-Scale and High-Order N-grams

郷津　優介[*1]　高野　渉[*1]　中村　仁彦[*1]
Yusuke Goutsu　Wataru Takano　Yoshihiko Nakamura

[*1]東京大学大学院　情報理工学系研究科　知能機械情報学専攻
Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo

Motion recognition is an essential technology for social robots in various environments such as homes, offices and shopping center, where the robots are expected to understand human behavior and interact with them. In this paper, we proposed the system composed of three models: motion language model, natural language model and integration inference model, and achieved to generate natural motion sentences with using large high-order N-gram. We confirmed not only that using more high-order N-gram improves a precision of generating motion sentences in the case of long sentence but also that the computational complexity of the proposed system is almost same as conventional one. In addition, we improved the precision by aligning the graph structure representing generated motion sentences into confusion network form. This means that simplifying and compacting morpheme word sequence have effect on the precision of generating motion sentences, too.

1.　はじめに

身の回りの環境には，画像・音声・行動などの膨大な情報が溢れている．これらの情報から予測を通じて新たな情報を想起するためには，実世界情報の記号化・言語化・言語構造化（文章化）などの操作が必要になる．人間の知能が他の動物と異なる点は，実世界にある膨大な情報を高度な記号である言語を通して理解でき，簡易な形態で操作することにより推論や連想などの知的な情報処理を行えることである．同様にして，ロボットが実世界における人間の行動を理解して行動支援を行うためには，身体的運動の記号化，運動記号の言語化，運動言語の言語構造化 [Takano 09] の技術が必要になり，言語推論・連想などを介した上でロボットが新しい運動・情報を生成することが求められる．本研究では，これらの機能を実現するために，従来の運動記号と単語の連想構造を表現する運動言語モデルに，大規模な高次 N-gram データを用いた単語間の並びを表現する自然言語モデルを統合した運動認識システムを提案する．そして，マルチドメイン・ロバスト・高精度な運動認識を行い，表現度の高い動作文生成の実現を目的としている．

2.　運動認識システムを構成する各種モデル

本研究で提案する運動認識システムは，図 1 に示すように運動記号の言語化を実現する運動言語モデル，運動言語の言語構造化を実現する自然言語モデル，これらのモデルから出力されるスコアにしたがって認識した運動を最尤に表現する動作説明文を決定する統合推論モデルの 3 モデルから構成される．

2.1　運動言語モデル

運動言語モデルに関しては，運動記号 λ，隠れ変数 s，単語 w の 3 層から構成され，λ から s が連想される確率 $P(s|\lambda)$ と s から w が連想される確率 $P(w|s)$ により結びついた構造となっている．隠れ変数の数は予め人手によって決められており，これら 2 つのモデルパラメータは EM アルゴリズムにより以下のように学習できる．

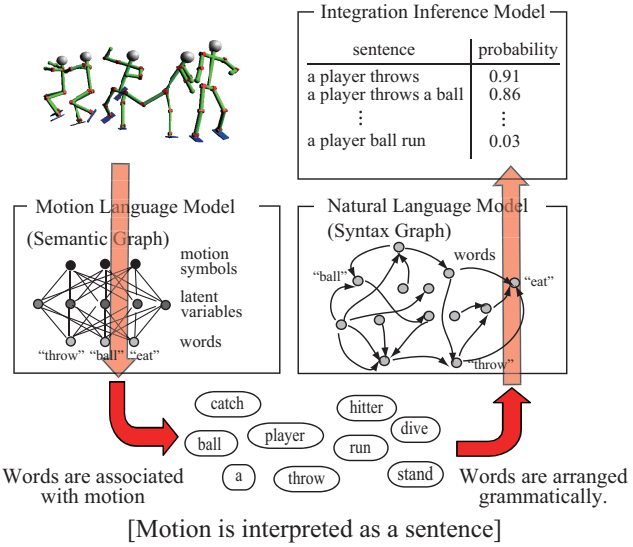

図 1: Overview of interpreting a motion as sentences (motion recognition system).

【E-step】

$$P(s|\lambda, w) = \frac{P(w|s)P(s|\lambda)}{\sum_s P(w|s)P(s|\lambda)} \tag{1}$$

として，隠れ変数の分布を運動言語モデルのパラメータから推定する．この推定された分布に基づき運動言語モデルのパラメータを以下のように最適化する．

【M-step】

$$P(s|\lambda) = \frac{\sum_{k=1}^{N}\sum_{i=1}^{n_k}\delta(\lambda,\lambda^k)P(s|\lambda^k,w_i^k)}{\sum_{j=1}^{N_s}\sum_{k=1}^{N}\sum_{i=1}^{n_k}\delta(\lambda,\lambda^k)P(s_j|\lambda^k,w_i^k)} \tag{2}$$

$$P(w|s) = \frac{\sum_{k=1}^{N}\sum_{i=1}^{n_k}\delta(w,w_i^k)P(s|\lambda^k,w_i^k)}{\sum_{j=1}^{N_w}\sum_{k=1}^{N}\sum_{i=1}^{n_k}\delta(w_j,w_i^k)P(s_j|\lambda^k,w_i^k)} \tag{3}$$

連絡先: 郷津優介，東京大学大学院情報理工学系研究科，〒113-8656 東京都文京区本郷 7-3-1，Tel: 03-5841-6381，Fax: 03-3818-0835，Email: goutsu@ynl.t.u-tokyo.ac.jp

Algorithm 1 calculating summation of word N-gram probabilities

```
len ← get the number of morpheme words in a sentence
totalLogP ← 0.0 {total of word N-gram probabilities}
for i = 1 to len do
    logP(i|i − len + 1, ..., i − 1) ← get the maximal word i-gram probability
    if logP(∗) is not log(0) then
        totalLogP ← totalLogP + logP(∗)
    end if
end for
```

式 (2) の右辺分子は，学習データ中において s が λ から生成された回数であり，分母はその正規化項である．式 (3) の右辺分子は，学習データ中において s が w から生成された回数であり，分母はその正規化項である．ここで，学習データは，ロボットが k 番目に観察した運動パターンの認識結果である運動記号 λ^k と，その運動パターンに人手で付与された正解文（単語列：$w_1^k, w_2^k, ..., w_{n_k}^k$）の対で構成される．$N$ は学習データの総数であり，N_s，N_w は隠れ変数と単語の種類の総数である．

2.2 自然言語モデル

多くの研究により様々な自然言語モデルが提案されてきている．本研究では，単純であるにも関わらず認識性能の改善が大きく，モデルのパラメータ推定が容易であるという理由で，単語 N-gram モデルを使って単語間の遷移を表現する文構造を規定する．一般的に単語 N-gram モデルは，1 次元の単語列 ($\mathbf{w} = \{w_1, w_2, \cdots, w_n\}$) における i 番目の単語 w_i の生起確率 $P(w_i)$ が w_i の直前の $(N-1)$ 単語に依存する $N-1$ 重マルコフ過程として以下のような式で近似的に表現される．

$$P(w_i|w_1, w_2, \cdots, w_{i-1}) \simeq P(w_i|w_{i-N+1}, \cdots, w_{i-1}) \qquad (4)$$

右辺の単語 N-gram 確率は，単語分割された学習用のテキストデータがあれば，そのテキストデータ中の単語列の相対頻度から推定することができる．

$$P(w_i|w_{i-N+1}, \cdots, w_{i-1}) = \frac{C(w_{i-N+1}, \cdots, w_i)}{C(w_{i-N+1}, \cdots, w_{i-1})} \qquad (5)$$

ここで，式 (5) の $C(w_{i-N+1}, \cdots, w_i)$ は単語列 $w_{i-N+1}, \cdots, w_i$ が出現する頻度を表す．自然言語モデルによる単語間の遷移確率は，式 (5) で推定される確率を用いて，ある注目ノードが遷移元の単語から遷移先の単語に到達するまでに計算した遷移毎の単語 N-gram 確率の総和になる．この単語間の遷移確率を決定するアルゴリズムを Algorithm 1 に示す．また，単語 N-gram 確率が計算できない場合は，バックオフ・スムージングにより推定された重みが単語 (N-1)-gram 確率に加算されるようになっている．このバックオフ・スムージングを含めた単語 N-gram 確率の計算アルゴリズムを Algorithm 2 に示す．

2.3 統合推論モデル

統合推論モデルでは，運動言語モデルと自然言語モデルの両スコアを用いて，運動記号からスコアの対数尤度が最大となる動作文（単語列：$w_1, w_2, ..., w_k$）を全体から探索する問題として表現される．ここで，運動記号 λ から動作文 $\mathbf{w}$ が生成される確率は以下のような式で計算される．

$$P(\mathbf{w}|\lambda) = \prod_{i=1}^{k} P(w_i|\lambda) \cdot \prod_{i=1}^{k} P(w_i|w_{i-N+1}, ..., w_{i-1}) \qquad (6)$$

Algorithm 2 finding the maximal word N-gram probability and accumulating backoff weights

```
initialization
repeat
    logP ← find log probability of context from trie node
    if logP is valid then
        record logP as the most specific one found so far
        reset backoffweight
    end if
    if i ≥ maximal context length or context[i] is none vocab then
        break
    end if
    next ← find context[i]
    if next is valid then
        accumulate backoffweight
        set next as next trie node
        increment i
    else
        break
    end if
until break command is occurred
return logP + backoffweight
```

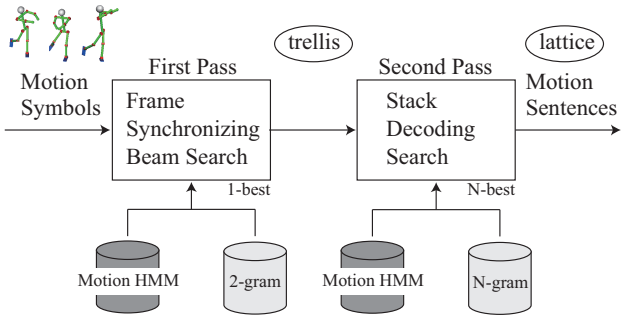

図 2: Outline of integration inference model. The whole search is divided into 2 steps.

この過程における運動認識アルゴリズムは，大規模で詳細な高次 N-gram を用いた自然言語モデルにおいても，精度を落とさずに効率良く認識を行えるように，全体は図 2 に示す 2 パスの段階的探索を行っている．まず，第 1 パスでは運動記号の入力に対して，運動言語モデル及び簡便な単語 2-gram モデルと最尤の直前単語からの Viterbi 経路のみを計算する 1-best 近似計算を用いて後向き (right-to-left) 探索による粗い認識を行う．この時に，各フレームにおいてビーム内に始端状態が残った単語候補を単語トレリス形式で保存する．単語トレリスとは，始終端フレームと入力終端から単語先頭までの累積スコアを保持した単語候補集合である．中間表現を単語トレリス形式にしておくことで，第 2 パスにおける探索空間を限定すると共に，A*アルゴリズムにおける先読み情報（未探索部分のヒューリスティック）として利用できる．これにより第 2 パスでは第 1 パスの結果を参照しながら，必要な部分にだけ精密な再計算を行える．第 2 パスでは，運動言語モデルと単語 6-gram までのモデルを用いて，前向き (left-to-right) のビーム幅付き best-first なスタックデコーディング探索を行う．この時に，第 1 パスでのスコアに基づいてフレーム毎に枝刈りの閾値を設定することで効率的に仮説を再評価する．この一連の

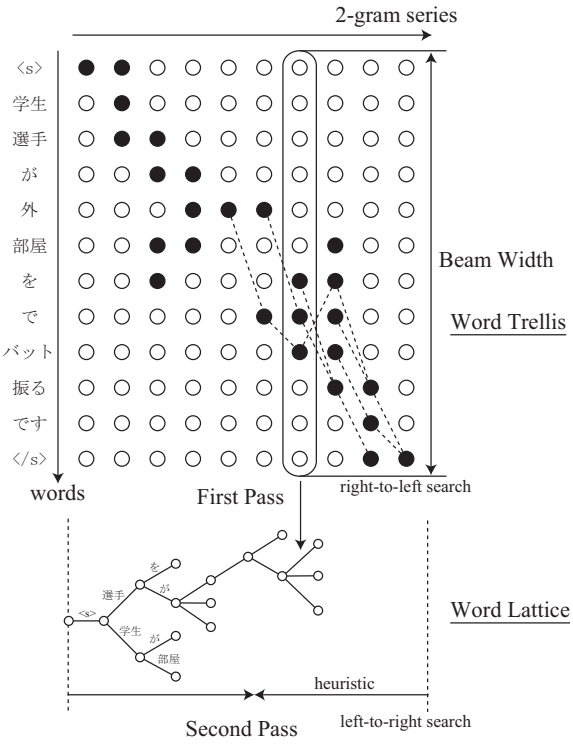

図 3: Word trellis and its use in word expansion on word lattice.

流れを示したものが図 3 である．しかし，最尤解を求める場合に，第 1 パスと第 2 パスでは N-gram モデルが異なり，後者が高いスコアを与える可能性があるために A*適格性を満たしておらず，最初に得られる解が最尤の動作文である保証はない．このため N 個の動作文候補をスコアでソートした N-best 探索を行い，最終的に生成される動作文を N-best リスト形式で出力している．ここで，N-best リスト形式をシンプル・コンパクトに表現したグラフ構造として，単語ラティス形式と 2.4 節で説明する形式が挙げられる．単語ラティス形式は，N を非常に大きく取った時に同一単語の集約などにより，N-best リスト形式を効率的に圧縮することができ，単語トレリス形式と異なりアークを単語としている．

2.4 コンフュージョンネットワーク

統合推論モデルからは単語ラティス形式ではなく，N-best リスト形式の状態で結果が出力される．ここで，複数の動作文から作成された単語系列がシンプル・コンパクトになるように，これらを表現するグラフ構造を単語コンフュージョンネットワーク (Confusion Network:CN) [Mangu 00] に変換（以後、アライメントと呼ぶ）する．また，単語 CN 形式は，2.3 節で説明した単語ラティス形式をさらにシンプル・コンパクトにしたものである．グラフ構造が N-best リスト形式から単語 CN 形式へアライメントされる様子を図 4 に示す．また，本研究におけるアライメントは以下の手順で行われる．

1. ＜ 上位候補のグラフ圧縮 ＞
   N-best リスト中の上位候補から単語ラティスを形成する．

2. ＜ 事後確率の計算 ＞
   単語ラティス上の各アークの事後確率を単語 2-gram スコアから Forward-Backward アルゴリズムで計算する．この時，単語ラティスには規定された単語系列のみが登録

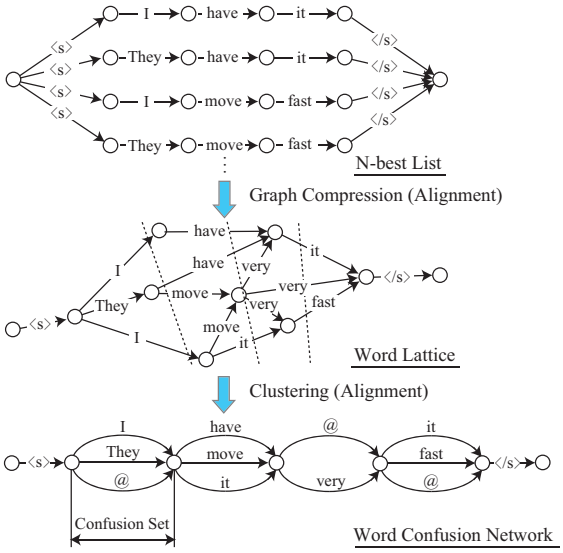

図 4: The alignment of graph structure from N-best list to word confusion network through word lattice.

されるため，多くの冗長な単語を含むだけでなく，入力文は規定された単語系列としか照合できない欠点がある．

3. ＜ 単語内クラスタリング ＞
   時間的なまとまりになるように，単語ラティス上で時間的重なりのある同一単語のアークをクラスタリングする．これによりできた時間的に競合している候補単語群をコンフュージョンセット (Confusion Set:CS) と呼ぶ．CS 上の各単語は存在確率の高い順にソートされている．

4. ＜“@”（null 候補）の追加 ＞
   クラスタリングにより，元の単語ラティスにおいて時間的に競合している単語が他のクラスにマージされた場合，CS 内の事後確率の総和が 1 より小さくなってしまう．このため任意の確率を持たせた “@”を追加して，CS の事後確率の総和が 1 になるように調整する．“@”では，その CS に単語が存在しないとしてスキップされる．

単語 CN 上では，単語ラティスに不可能だった単語系列まで探索できるようになる．また，アライメントの際にクラスタリングが行われるが，元の単語ラティス上の単語の順序関係は保持され，マージ後のクラスにおいても言語制約は適応される．

3. 実験結果

運動言語モデルの学習データには，467 種類の運動記号に合計 764 個の動作説明文を人手で付与した行動コーパスを用いた．運動記号は，被験者に貼り付けた 35 個のマーカーの位置を光学式モーションキャプチャにより計測して，計測した運動パターンデータを HMM により記号化したものである．また，自然言語モデルの学習データには，マルチドメイン，ロバスト，高精度なモデルを構築できるように Google N-gram コーパス（正式名称：Web 日本語 N グラム）を用いた．Google N-gram コーパスは一般に公開されている日本語の Web ページにおいて Google がクロールしたものから抽出されている．抽出対象となったデータは約 200 億文（約 2550 億文単語）で，出現頻度が 20 回以上の 1–7-gram までが収録されている．実際に使

表 1: Cut-off and Total number of each N-gram

N	Cut-off	Total Number	
		Before Cut-off	After Cut-off
1-gram	50,000	2,565,424	67,260
2-gram	5,000	80,513,289	3,769,894
3-gram	1,000	394,482,216	17,593,003
4-gram	1,000	707,787,333	20,132,262
5-gram	800	776,378,943	19,485,755
6-gram	500	688,782,933	18,521,684

表 2: Processing time of two motion recognition systems

Motion Index		60	260	290	329	386
Words		3	3	3	4	3
Time[s]	Conventional	5.46	7.13	5.49	16.80	9.16
	Proposed	8.00	6.68	6.27	49.53	7.82

用する N-gram の次数は 6 までとし，消費するメモリ容量の関係で出現頻度によるカットオフを行っている（Google N-gram コーパスの全ての語彙を使用すると数 100GB のメモリが必要になる）．ここで，収録された N-gram やカットオフの詳細な情報について表 1 に示す．ただし，After cut-off 欄では未知語処理を行っており，1-gram で作成された単語ファイルに含まれない N-gram($N > 1$) はカウントしていない．

統合推論モデルの段階的探索における単語トレリスの作成には，ここでの単語 2-gram データを使用した．また，対数尤度の最大値は，式 (6) からも分かるように運動言語モデルと自然言語モデルのスコアを加算した総合スコアから求めており，両者の重みは同じにしている．

統合推論モデルの評価には，生成された動作文と正解文の単語単位の DP マッチングにより編集距離を正規化した単語誤り率 (Word Error Rate:WER) を用いた．編集距離とは，置換 (Sub)，挿入 (Ins)，削除 (Del) 誤りの割合のことである．

$$\begin{aligned} WER &= \frac{\alpha_S \cdot Sub + \alpha_D \cdot Del + \alpha_I \cdot Ins}{Words} \\ &= Sub_{score} + Del_{score} + Ins_{socre} \end{aligned} \quad (7)$$

ここで，$Words$ は正解文の単語数，Sub は置換，Ins は挿入，Del は削除の操作数を表す．また，それぞれに対して α_S，α_I，α_D の比率で重みを付与した．この値が小さいほど性能が良い．

ただし，本研究で使用した PC の動作環境は，Dell Precision T7500 (CPU:2.40GHz, メモリ:48GB, OS:64bit) である．

3.1 運動認識システムによる動作文生成

学習データに 3 単語の動作文が付与された運動パターンの中からランダムに選択した 5 個に対して，運動認識システムによる動作文生成の実験を行った．表 2 には従来システム [Takano 09] と提案システムのそれぞれの処理時間を示した．ここで，N-gram の次数とビーム幅は，3 単語の動作文に対しては 3, 25 に，4 単語の動作文に対しては 4, 45 とした．統合推論モデルにおける 2 パスの段階的探索により，従来システムよりも高次の N-gram まで探索に用いているにも関わらず，3 単語の動作文生成に関しては平均で 0.4s の違いしか見られなかった．

3.2 N-gram の次数を変えた時の処理時間と WER

学習データに 5 単語の動作文が付与された運動パターンの中からランダムに選択した 5 個に対して，N-gram の次数を 2–6 に変化させた時の処理時間と WER の平均値を比較した．表 3 の各運動パターンの WER は，単語数が 5 以上の N-best リストから抽出した上位 10 個の平均値を表している．ここで，Sub，Ins，Del に対する重みは $\alpha_S : \alpha_I : \alpha_D = 2 : 3 : 3$ とした．また，ビーム幅は 70 とした．単語 2-gram モデルと単語 6-gram モデルの精度を比較すると，WER が約 20.3%減少していることが分かる．

表 3: Time and WER when varying the order of N-gram

N	Time[s]	Sub	Ins	Del	WER	Words
2-gram	7.7	0.22	0.54	0.54	3.84	5
3-gram	56.4	0.22	0.48	0.48	3.33	5
4-gram	227.2	0.14	0.54	0.54	3.50	5
5-gram	417.4	0.43	0.43	0.43	3.44	5
6-gram	618.7	0.36	0.39	0.39	3.06	5

表 4: WER of 10-best average, 1-best and CN-best

	Sub	Ins	Del	WER	Words
10-best Avg	0.36	0.39	0.39	3.06	5
1-best	0.40	0.32	0.32	2.72	5
CN-best	0.44	0.16	0.16	1.84	5

3.3 単語 CN 形式に変換した時の処理時間と WER

3.2 節と同じ 5 個の運動パターンに対して，N-best リスト形式から単語 CN 形式にアライメントした時の WER を比較した．表 4 の 10-best Avg は単語数が 5 以上の N-best リストから抽出した上位 10 個の WER の平均値を，1-best はその中の最尤候補文の WER を用いた．また，CN-best とは単語数が 5 以上となる範囲で CS 列を走査していった時に WER が最小となる値を用いた．ここで，重みに関しては 3.2 節と同じにした．また，N-gram の次数とビーム幅は 6, 70 とした．1-best と CN-best の精度を比較すると，WER が約 32.4%減少していることが分かる．

4. 結言

ヒューマノイドロボットが実世界の運動情報を言語的に理解して，実世界に作用するための知能設計論として，3 つのモデル（運動言語モデル，自然言語モデル，統合推論モデル）からなる運動認識システムを提案し，従来のシステムでは不可能であった大規模な高次 N-gram を用いた自然な動作文生成が実現できることを示唆した．また，長文における運動認識の精度を向上させるために，複数の単語系列を表現するグラフ構造を単語 CN 形式に変換して，コンパクト／シンプル化による精度の改善を実現した．

本研究は，科研費基盤研究 (S)（20220001, 代表者：中村仁彦）並びに，平成 24 年度文部科学省科学研究費補助金挑戦的萌芽研究（代表者：高野渉）の支援を受けて行った．

参考文献

[Mangu 00] Mangu, L., Brill, E., and Stolcke, A.: Finding consensus in speech recognition: word error minimization and other applications of confusion networks, *Computer Speech & Language*, Vol. 14, No. 4, pp. 373–400 (2000)

[Takano 09] Takano, W. and Nakamura, Y.: Incremental learning of integrated semiotics based on linguistic and behavioral symbols, in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pp. 2545–2550IEEE (2009)