社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2006－ＦＩ－ 84 (2)
2006－ＮＬ－175 (2)
2006／9／12

道案内インストラクションからの知識Frame抽出

清水伸幸　Andrew Haas

コンピュータサイエンス学科
ニューヨーク州立大学アルバニー校

要旨

アシスタント・ロボットを作るには、自然言語で表現された人間のインストラクションを解釈し実行する必要がある。　この目的のため、本論文ではオフィス空間を想定した道案内のインストラクションから、道順をたどるのに必要な知識Frameを機械学習を用いて抽出する。　方法としては、インストラクションを行動毎にセグメントし、知識Frameにマップするjoint modelとなるframe-segment decoding algorithmを提唱し、perceptronで学習する。　同時に、知識Frameのslot fillerを一つにまとめて行動ごとにlabelを作り、linear-chain Conditional Random Fields (CRFs)を応用可能にした上で、これと比較する。実験結果では、frame-segment modelが77.7%の成功率でCRFsを上回った。

Knowledge Frame Extraction From Navigational Route Instruction

Nobuyuki Shimizu　Andrew Haas

Computer Science Department
State University of New York at Albany

Abstract

To build a simulated robot that follows route instructions in unconstrained natural language, we propose a frame-segment decoding algorithm that achieves two joint tasks: (1) Segment the route instruction so that the sequence of segments corresponds to a sequence of actions required to complete the task, and (2) Choose the slot fillers of the frame-based knowledge representation for each action. Our model was trained with voted perceptron. We also created labels for actions by merging the slot fillers of their frame-based knowledge representation, reducing the problem to a sequence labeling task. As a comparison, we applied a linear chain Conditional Random Fields to this representation. The experimental result shows that our model performs better with a 77.7% success rate.

1 始めに

コンピュータ、ロボットが今まで以上に人間の生活の一部として機能するためには、使い手に学習を強いない自然言語によるインターフェイスが重要である。　この目的を達成する一段階として、命令

構文などの制約のない、オフィス空間での道案内のインストラクションを機械学習の手法を用いて解釈する。　以下は実際に学部生が書いたインストラクションの例である。

" Head straight then make a right turn, then head straight again ignoring the first left and right rooms, but not ignoring the second room to the right, enter it."

" walk straight down the hallway to where the hallway t' s and bear to the left. as soon as you bear left the hallway breaks again and bears to the right. take that immediate right and continue straight down the hallway. enter into the doorway directly in front of you at the end of the hall."

これらの例から、オフィス空間での道案内インストラクション解釈という限られたドメインであるにもかかわらず、この問題が容易に解けないことが分かる。

2 関連文献

[Shimizu, 2006]に付け加え、このドメインの過去の研究にはWinograd [1972]によるSHRDLU program、[Lauria et al., 2001; 2002]で発表されたIBL (Instruction-based Learning for Mobile Robots) project などがある。　しかし上記の二つはいずれも、制約のない自然言語から学習し、解釈するシステムではなく、成功率も発表されていない。　他の situated language experiments の試みには、[MacMahon and Stankiewicz, 2006; Stoia et al., 2006]が含まれる。　本稿と同じく、道案内を解釈し、目的地到着成功率を調べた例としては、[MacMahon et al., 2006]がある。　MacMahonらは、本稿より伝統的なparse treeを作り、そこからframeを取り出すというアプローチで問題に取り組んだ。　本稿では、[Shimizu, 2006]での問題定義とコーパスを引継ぎ、そこで使われたCRFs [Lafferty et al. , 2001] によるによるタスク成功率を凌ぐ学習モデルを提案する。

3 タスク

第一に、このタスクにおけるインプットとアウトプットについて述べる。　オフィス空間での道案内インストラクションを解釈するためには、三つのインプットが必須となる。

- 制約のない英語で書かれたオフィスに行くためのインストラクション。
- オフィス空間の知識。
- エージェントの場所と向き。

アウトプットはインストラクションでエージェントが行くべきオフィスの場所である。　この問題は二つのコンポーネントに分解できる。

- インストラクション → セグメント → 行動シークエンス
- 行動シークエンス * オフィス空間 * 位置と向き → 目的地

前者は情報抽出システム、後者は行動シークエンスを現在地と地図に照らし合わせ、目的地を見つけるシステムである。　本稿では前者の情報抽出システムを改良する。　後者のシステムを前者の抽出システムと統合する方法、あるいはデータの傾向については[Shimizu, 2006]を参照のこと。

第二に、知識Frameについて説明する。　このドメインにおいては、ランドマークとは廊下かドアである。　行動とは、ランドマークにたいして行われる三つまでの動作となる。　可能な動作とは：ランドマークまで前進する（advance）、ランドマークの方向に合わせて右か左を向く（turn）、ドアであれば中に入る（enter）、の三つとなる。　動作は常にこの順番で行われ、一つの行動となる動作の組み合わせは[advance,turn], [advance,turn,enter], [advance,enter]である。

この行動は行動Frameとして、四つのdimensionがあるvectorとして表現される。 ここでは、便宜的にdimensionをslot、そこに入る特定のオブジェクトをslot fillerと呼ぶ。 一つ目のslotには、この行動がenterを含むのか、それとも行くだけ(go)なのかを区別するG,Eのいずれかが入り、二つ目のslotには廊下(hall)あるいはドア(door)を示すH,Dのいずれかが指定される。 三番目には右(R)、左(L)、真直(S)のいずれかが入り、四番目のslotには、一番目、二番目、三番目、あるいは、最後のランドマークを示す1,2,3,Zのいずれかが入る。 〈(G|E),(H|D),(L|R|S),(1|2|3|Z)〉 したがって、〈G,H,R,2〉 は「右側の二番目の廊下まで前進し、廊下側（右側）を向く」という意味であり、〈E,D,L,3〉 は「左側の三番目のドアの中に入る」という行動を表す。 我々が用いたオフィス空間では15の行動がありえることとなる。 本稿では、行動Frameでラベル付けされた、[Shimizu, 2006]と同じ427個のインストラクションを用いて学習と評価を行う。

3 学習方法

[Shimizu, 2006]における linear chain CRFs の問題点は、〈G,H,R,2〉 などの行動Frameを一つのアトミックなラベルと捉えた後、BIO tagging format [Ramshaw and Marcus, 1995]を用いて、品詞付けに類似する単語のラベル付け問題として解いた点である。 このため、" take the first door on the left" や、" take the second door on the left" といった、単語一つしか違わないセグメントにおいても、全く違ったラベルが全ての単語に与えられる。 このケースでいえば、前者は全ての単語が EDL1、後者は全ての単語が EDL2 でラベル付けされる。 コーパス内では EDL1 の出現頻度が EDL2 よりも高いため、後者が EDL1 と解釈されてしまうケースが多く見られた。 理想的には、" take X door" の句で 一番目の G,E を決定し、" first"," second" などの単語により、四番目の 1,2,3,Zのいずれかを決定できるよう、ラベルが分解されるべきであり、[Cohen and Sarawagi, 2004; Sarawagi and Cohen, 2004]を拡張することでこの目的を達成する。

[Cohen and Sarawagi, 2004]では、マーコフ仮定の元、全てのセグメントのシークエンスから、最もスコアが高い組み合わせを発見するDynamic Programmingの手法が使われている。 つまり、例として、" take left, second door on right" があるとすると：

セグメントが一つだけのケース、
[take left, second door on right]、
セグメントが二つあるケース、
[take][left, second door on right]から、[take left, second door on][right]、
セグメントの数を増やしていき、最終的には全ての単語がセグメントのケース、
[take][left][,][second][door][on][right]

といったセグメントのシークエンスが有り得る。 マーコフ仮定（一つのセグメントのスコアは左右にある直近のセグメント以外から影響を受けない）があれば、Complexityは$O(n^2)$である。（n:=インプットの長さ） 我々の拡張は以下のようなものである。 Dynamic Programming中、セグメントを一つづつとり、まず、一番目のslotにはいる E でセグメントの単語全てをラベルづけしたとしてスコアを出す。 そして、同じようにして得られた G のスコアと比べ、いずれか高い方を同じセグメントのスコアに足す。 これをslotの数だけ繰り返すことで、Frameとしてのセグメントのスコアを出し、上記の手法と組み合わせて、Complexityを増やすことなく最もスコアの高いシークエンスのスコアを発見し、バックポインタを用いて、セグメントとslot fillerの部品を取り出すことで、行動Frameを取り出す。 学習には[Cohen and Sarawagi, 2004]と同じく、[Collins, 2002]による

averaged perceptron を用いる。　具体的には、行動Frameのシークエンスの中で、間違った部品が入っていれば、その部品のスコアを下げ、入っているべき部品のスコアを上げる。　これを繰り返すことで学習が行われる。

```
Decoding Algorithm : DECODE(the scoring function s(p))
score := 0;
for q := index_start to index_end
        for length := 1 to index_end - q
                r := q + length;
                frame_score := 0;
                for each slot S
                        slot_score := 0;
                        for each filler P for slot S
                                score := 0;
                                if (length > 1)
                                        score := score + s(<P,P,r-2,r-1>);
                                        # 上記は隣の slot filler も同じである場合に足すスコアを足す。
                                        # r-2, r-1は、ラベルづけする単語のポジションである。
                                        score := score + filler_table[q][r-1][P];
                                score := score + s(<P,r-1>);
                                # 上記は r-1 のポジションにある単語が slot filler P で
                                # ラベル付けされた時のスコアを足している。
                                if (score >= filler_table[q][r][P])
                                        filler_table[q][r][P] := score;
                                if (score >= slot_score)
                                        slot_score := score;
                        end for
                        frame_score := frame_score + slot_score;
                end for
                frame_score := frame_score + s(<I,q,r-1>);
                # 上記はセグメントがポジション q から始まり、ポジション r-1 で終わる場合
                # のスコアを足すものである。
                if (index_start < q)
                        frame_score := frame_score + s(<I,I,q-1, q>);
                        # 上記は、前のセグメントがポジション q-1 で終わり、次のセグメントが
                        # ポジション q から始まる場合のスコアを足している。
                        frame_score := frame_score + segment_table[q];
                if (frame_score >= segment_table[r])
                        segment_table[r] := frame_score;
        end for
end for
return (segment_table[index_end])
```

Note: Since the scoring function s(p) is defined as w f (x_i, {p}), the input sequence x_i and the weight vector w are also the inputs to the algorithm.

行動Frameシークエンスの部品として使われるものは次のようなものである。 まず、slot fillerのtransition、〈P,P,r-2,r-1〉。 これは、同じslot filler P が、隣同士にあり、セグメントの切れ目が間にないことを示している。 r-2 と、r-1 は、slot filler P がラベルづけするインストラクション内の単語の位置である。 次に、slot fillerのstate、〈P,r-1〉。 これは、r-1 番目の単語が slot filler P でラベル付けされていることを示す。 セグメントの部品としては二つあり、まず、セグメントのtransition、〈I,I,q-1, q〉 は、セグメントの切れ目が q-1と、qの間にあることを示す。 セグメントのstate、〈I,q,r-1〉 は、セグメントが q の位置から始まり、r-1 の位置で終わることを示している。 これらの部品と、r-1など、インデックスで示された位置の周りにあるインストラクション内の単語との組み合わせを素性とし、素性毎に学習された重みを足すことでスコアの合計が決まる。 部品のスコアは s(x) = w f (x, p) の関数で表される。 w は weight vector, f(・,・)は上記の単語と部品の組み合わせの存在を表すvector-valued feature function、x がインストラクション、p が部品となる。

4 結果

6 way cross fold validationの結果、目的地到達成功率は、本稿で提唱したモデルでは77%に達した。同じ条件で実験が行われた、[Shimizu, 2006]の linear-chain CRFs の 73%と比べ、大きな成功率の向上が見られた。 残り20%ほどの道案内インストラクションは、始めの例であげられたように難しいケースが多く、本稿の問題設定ではいかなるアルゴリズムでも困難が予想される。 さらなる性能の向上にはlabelingやsemantic representationの工夫、co-referenceや、インストラクション内で目的達成に必要ない部分を認識する技術、より良い最適化のアルゴリズムなどが求められる。

References

[Cohen and Sarawagi, 2004] W. Cohen and S. Sarawagi. Exploiting dictionaries in named entity extraction: Combining semi-markov extraction processes and data integration methods. In Proc. of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and DataMining, 2004.

[Collins, 2002] M. Collins. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In Proc. of Empirical Methods in Natural Language Processing (EMNLP) , 2002.

[Lafferty et al. , 2001] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Proc. of the 18th International Conference on Machine Learning , 2001.

[Lauria et al. , 2001] S. Lauria, G. Bugmann, T. Kyriacou, J. Bos, and E. Klein. Personal robot training via naturallanguage instructions. In IEEE Intelligent Systems , volume 16:3, pages 38-45, 2001.

[Lauria et al. , 2002] S. Lauria, T. Kyriacou, G. Bugmann, J. Bos, and E. Klein. Converting natural language route instructions into robot-executable procedures. In Proc. of the 2002 IEEE Int. Workshop on Robot and Human Interactive Communication , pages 223-228, 2002.

[MacMahon and Stankiewicz, 2006] Matt MacMahon and Brian Stankiewicz. Human and automated indoor route

instruction following. In Proceedings of the 28th Annual Conference of the Cognitive Science Society , Vancouver, BC, July 2006.

[MacMahon et al. , 2006] Matt MacMahon, Brian Stankiewicz, and Benjamin Kuipers. Walk the talk: Connecting language, knowledge, and action in route instructions. In Proceedings of the 21st National Conf. on Artificial Intelligence (AAAI-2006) , Boston, MA, July 2006.

[Malouf, 2002] R. Malouf. A comparison of algorithms for maximum entropy parameter estimation. In Proc. of the Sixth Conf. on Computational Natural Language Learning (CoNLL) , 2002.

[Manning and Schutze, 1999] C. Manning and H. Schutze. Foundations of Statistical Natural Language Processing. MIT Press, Cambridge,Massachusetts, 1999.

[McCallum et al. , 2003] A. McCallum, K. Rohanimanesh, and C. Sutton. Dynamic conditional random fields for jointly labeling multiple sequences. In Workshop on Syntax, Semantics, Statistics. (NIPS) , 2003.

[Peng and McCallum, 2004] F. Peng and A. McCallum. Accurate information extraction from research papers using conditional random fields. In Proc. of the Human Language Technology Conf. (HLT), 2004.

[Ramshaw and Marcus, 1995] L. Ramshaw and M. Marcus. Text chunking using transformation-based learning. In Proc. of Third Workshop on Very Large Corpora. ACL,1995.

[Sarawagi and Cohen, 2004] S. Sarawagi and W. Cohen. Semi-markov conditional random fields for information extraction. In Advances in Neural Information Processing Systems 16 , 2004.

[Sha and Pereira, 2003] F. Sha and F. Pereira. Shallow parsing with conditional random fields. In Proc. of the Human Language Technology Conf. (HLT) , 2003.

[Shimizu, 2006] N. Shimizu. Semantic discourse segmentation and labeling for route instructions. In Proc. of the 44th AnnualMeeting of the ACL, Student ResearchWorkshop, 2006.

[Stoia et al. , 2006] L. Stoia, D. Byron, D. Shockley, and E. Fosler-Lussier. Sentence planning for realtime navigational instruction. In Proc. of the Human Language Technology Conf. (HLT) , 2006.

[Winograd, 1972] T. Winograd. Understanding Natural Language . Academic Press, 1972.