ACL 2012 参加報告

NLP若手の会 第7回シンポジウム
2012年9月4日

徳永拓之
(株)Preferred Infrastructure
tkng@preferred.jp
http://twitter.com/tkng

ACL2012

- 本会議：7/10～7/12の三日間
- 場所：韓国 済州島

来年以降のACL

- ACL2013：ブルガリア
- ACL2014：ボルチモア（アメリカ）
- ACL2015：場所未定(アジア)　IJCNLPと共催

アジェンダ

- 招待講演 (2P)
- Lifetime Achivement Award (1P)
- その他面白かった発表の紹介

Best Paper Awardについてはご本人の発表があるので割愛します

非常に偏った報告となることをあらかじめお詫び申し上げます

招待講演1: Remembrance of ACLs past

by Aravind K. Joshi

- 50年のACLの歴史をふり返る
- AMTCL (MT=機械翻訳) という名前で始まった
- 温故知新は大事である
  - ツリートランスデューサを使った研究はメモリを使い過ぎるので一旦廃れたが、復活した
- 言語学ともっと交流すべき

招待講演2:
Computational linguistics: Where do we go from here?

by Mark Johnson

- CLをScienceにしたい
  - Fourier変換みたいに、工学先行の分野はよくある
  - language acquisitionとかneurolinguisticsとかをもっと流行らせたい
- CLは他の分野に溶け込んでなくなるかも

Lifetime Achivement Award

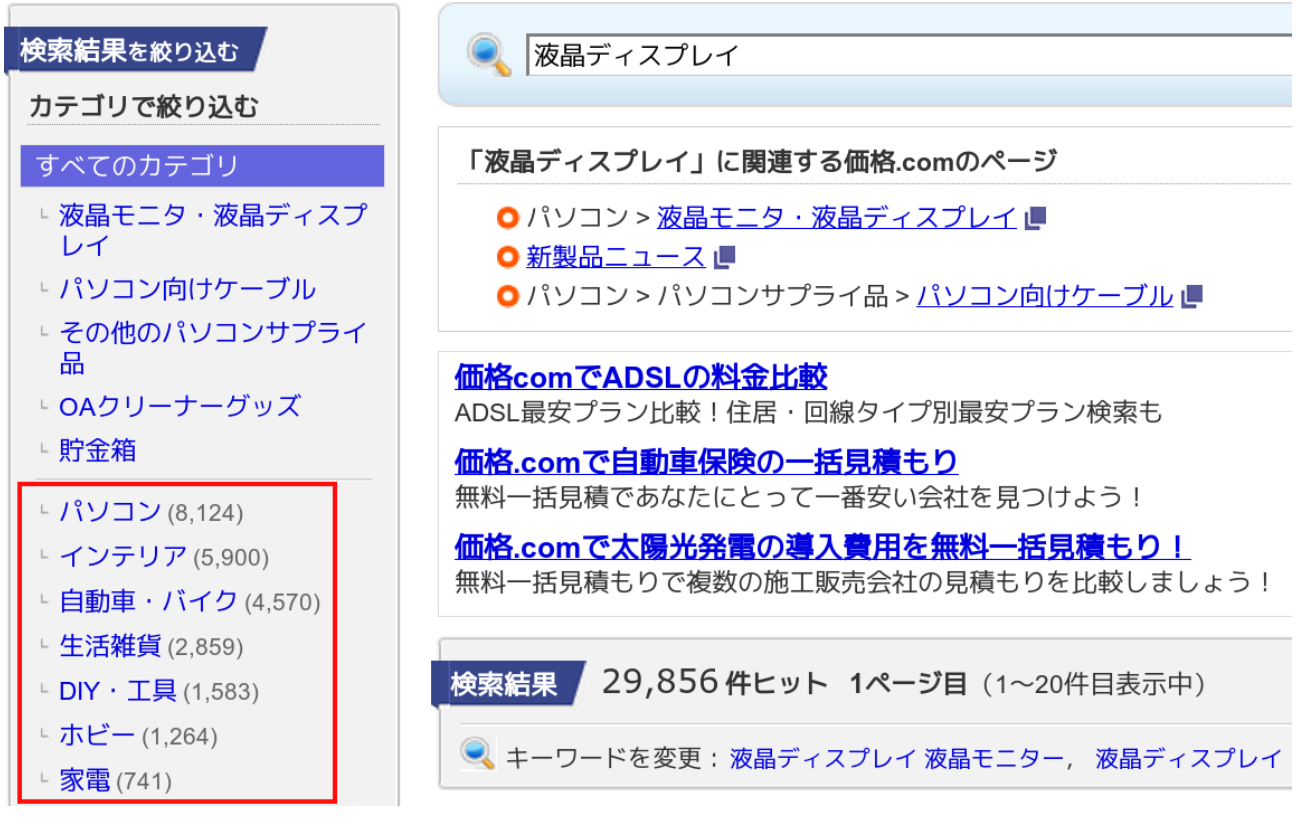
- Charles Fillmore, FrameNetの主導者
- Chomskyが出てきて全てが変わったと話していたことが印象的
- 彼の半生が語られたが、引っ越しするたびに引越し先の冬の平均気温がいちいち提示され、シュールな笑いを生んでいた

ここからは普通の発表の話に移ります
4件の発表について紹介します

- Structuring E-Commerce Inventory
- Computational Approaches to Sentence Completion
- Baselines and Bigrams: Simple, Good Sentiment and Topic Classification
- Spectral Learning of Latent-Variable PCFGs

Structuring E-Commerce Inventory [Mauge+] (1/2)

- 属性表の自動生成のため、属性名を抽出する
  - 属性表はファセットサーチなどに使える有用なデータである



Structuring E-Commerce Inventory [Mauge+] (2/2)

- 4つのパターンで属性名がだいたい網羅できる
  - [NP][:][optional DT][NP] みたいな感じ
  - データセットが大きいからこれでいけるんだぜ、とのこと
- 「作者」と「著者」みたいに同一の属性名を後で分類器に入れてくっつける
  - 同一文書中に出てくるかとかが素性として効く
- 全体的に工夫が効いており賢い

Computational Approaches to Sentence Completion [Zweig+] (1/2)

- MSRとCornell大学、UCIの人々
- SAT形式の穴埋め問題を解く

1. One of the characters in Milton Murayama's novel is considered _____ because he deliberately defies an oppressive hierarchical society.
(A) rebellious (B) impulsive (C) artistic (D) industrious (E) tyrannical

Computational Approaches to Sentence Completion [Zweig+] (2/2)

- NIIの人口頭脳プロジェクト（ロボットは東大に入れるか）と解いてる問題が似ている
- データセットを公開しているのはポイントが高い
- 問題の解き方自体はベーシック
  - LSAやN-gramを使う

Baselines and Bigrams: Simple, Good Sentiment and Topic Classification [Wang and Manning] (1/2)

- 評判やトピックの分類タスクを解く
- word bigram featureは評判分類に効く
- 文章が短い場合、Naive Bayes > SVM
- 文章が短い場合にも性能のよいNBSVMを提案

Baselines and Bigrams: Simple, Good Sentiment and Topic Classification [Wang and Manning] (2/2)

- NBSVMでは、SVMに投入するデータをMNBの式を使って予め変形する

the count vectors as $\mathbf{p} = \alpha + \sum_{i:y^{(i)}=1} \mathbf{f}^{(i)}$ and $\mathbf{q} = \alpha + \sum_{i:y^{(i)}=-1} \mathbf{f}^{(i)}$ for smoothing parameter α. The log-count ratio is:

$$\mathbf{r} = \log\left(\frac{\mathbf{p}/||\mathbf{p}||_1}{\mathbf{q}/||\mathbf{q}||_1}\right) \quad (2)$$

- NBSVMは他のタスクにも有用だろうか？

Spectral Learning of Latent-Variable PCFGs [Cohen+]

L-PCFGをSpectral Learningする

- Spectral Learningとはなにか？
- L-PCFGとはなにか？
- L-PCFGのSpectral Learningでの学習

Spectral Learningとは？

- ある行列に対する特異値(固有値)を使う学習手法に対する総称（とここでは定義します）
  - spectral decomposition = eigendecomposition
  - 調べた中では、spectral decompositionを使っている手法よりもSingular Value Decomposition (SVD)を使っている手法の方が多い
- どんな行列になるかは解きたい問題次第

なぜSpectral Learningがよいか？

- ある条件のもとでは最適解が求まる
  - HMMのような局所解のある問題であっても！
  - 元と完全に同じ問題を解いてるかは謎
    - 誤差はこれぐらいに収まるよ、みたいな解析はある
- SVDは研究が進んできて、高速に解ける
  - RedSVDとか使うと自分で実装しなくていい
  - http://code.google.com/p/redsvd/

Spectral Learningの歴史

- 昔から存在したが、クラスタリング向けだった
- HsuらがObservable Representationという問題の変形法を導入し、隠れマルコフモデルに適用できることを示した（2009）
  - ここはかなり大きなポイント
- CohenらがHsuらの手法を拡張し、L-PCFGに適用した（←今ココ）

ここから一旦L-PCFGの話に移ります

L-PCFGとは

Latent Probabilistic Context Free Grammer

- 隠れ状態を持ったPCFG
- CFG → PCFG → L-PCFGの順で説明します

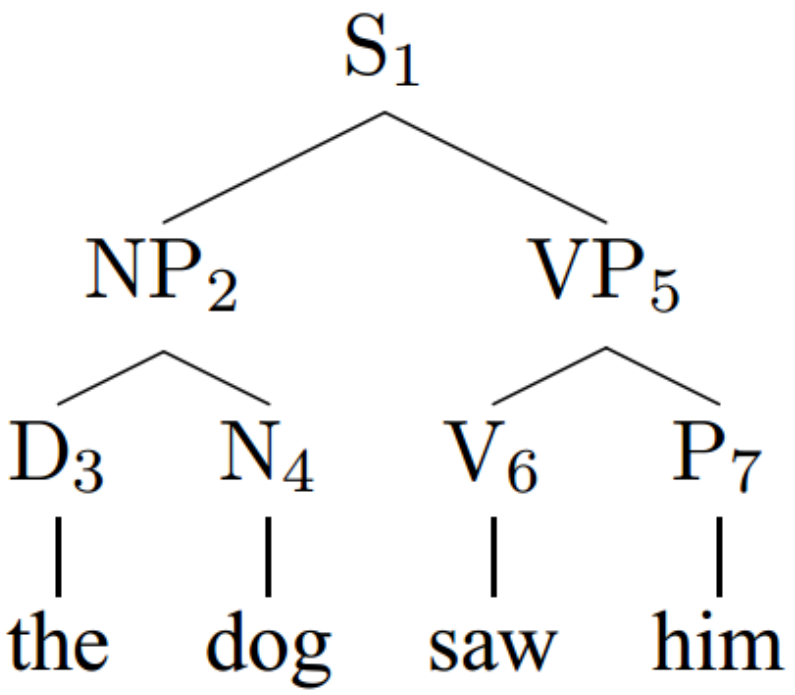
まず、CFGとは

以下の2つの形の導出ルールで文を作る文法

A, B, C は非終端記号、aは終端記号の例とします

- A → BC　（非終端記号の生成）
- A → a　（終端記号の生成）
  - 全ての規則がこの形をしている場合をチョムスキー標準形と呼ぶ
  - 全てのCFGはチョムスキー標準形に変換できる

CFGでの構文木の導出例



$r_1 = \text{S} \rightarrow \text{NP VP}$

$r_2 = \text{NP} \rightarrow \text{D N}$

$r_3 = \text{D} \rightarrow \text{the}$

$r_4 = \text{N} \rightarrow \text{dog}$

$r_5 = \text{VP} \rightarrow \text{V P}$

$r_6 = \text{V} \rightarrow \text{saw}$

$r_7 = \text{P} \rightarrow \text{him}$

Figure 1: An s-tree, and its sequence of rules. (For convenience we have numbered the nodes in the tree.)

CFGの問題点

ある文（終端記号列）に対して取り得る構文木の形が複数あり得る

- PCFGでは、それぞれの構文木に対して確率を付与することで、この問題に対処する

PCFGの問題点

- 非終端記号の粒度が荒い
  - かといって粒度を細かくするとアノテーションが辛くなる
  - 同じ非終端記号であっても、場合によってルールに与える確率を変えたい

- 隠れ状態を導入することで、上の問題に対処するのがL-PCFG

L-PCFGの定義

- 終端記号、非終端記号、生成ルール、ルールに対する確率によって定義される
- PCFGとの違い：生成ルールが隠れ状態というパラメーターを取るようになる

L-PCFGで一番重要な計算

- ツリー（構文木）に対して確率を計算する
- ツリー中の全てのルール（と全ての隠れ状態）に対して以下の計算を行い、その積をツリーに対する確率とする

$$p(a(h_1) \rightarrow b(h_2)\ c(h_3)|a(h_1)) =$$
$$q(a \rightarrow b\ c|h_1, a) \times s(h_2|h_1, a \rightarrow b\ c) \times t(h_3|h_1, a \rightarrow b\ c)$$

$$\text{and } p(a(h) \rightarrow x|a(h)) = q(a \rightarrow x|h, a).$$

提案手法の概要

前提：L-PCFGではすべての隠れ状態について総和を取るという操作が必要となる

- 提案手法ではまず、素敵なパラメーターテンソルC（とその他いくつかのパラメーター）が存在すると仮定する
  - このCがあれば主要な計算が実行できるものとする
- Observable representationという、隠れ状態を考慮せずに計算できる統計量を考える
  - この前準備としてSVDが出てくる
- Observable representationからテンソルCを推定する

テンソルが出てきてしまった…

```
＿人人人人人人人人人＿
＞　突然のテンソル　＜
￣^Y^Y^Y^Y^Y^Y^Y^Y^￣
```

テンソルという名前は付いているが

- 以下で出てくるテンソルは単なる3次元配列と思ってよい (Tucker decompositionとかは出てこない)
- テンソルは以下の形でしか出てこない

Definition 1 *A tensor* $C \in \mathbb{R}^{(m \times m \times m)}$ *is a set of* m^3 *parameters* $C_{i,j,k}$ *for* $i, j, k \in [m]$. *Given a tensor* C, *and a vector* $y \in \mathbb{R}^m$, *we define* $C(y)$ *to be the* $(m \times m)$ *matrix with components* $[C(y)]_{i,j} = \sum_{k \in [m]} C_{i,j,k} y_k$.

Finally, for vectors $x, y, z \in \mathbb{R}^m$, $xy^\top z^\top$ is the tensor $D \in \mathbb{R}^{m \times m \times m}$ where $D_{j,k,l} = x_j y_k z_l$ (this is analogous to the outer product: $[xy^\top]_{j,k} = x_j y_k$).

テンソルを用いた場合のツリーに対する確率計算

Algorithm: (calculate the f^i terms bottom-up in the tree)

- For all $i \in [N]$ such that $a_i \in \mathcal{P}$, $f^i = c^{\infty}_{r_i}$
- For all $i \in [N]$ such that $a_i \in \mathcal{I}$, $f^i = f^{\gamma} C^{r_i}(f^{\beta})$ where β is the index of the left child of node i in the tree, and γ is the index of the right child.

Return: $f^1 c^1_{a_1} = p(r_1 \dots r_N)$

出てきそうな疑問

- そんな素敵なCが存在するの？
  - ある条件を満たせば存在します (次ページ)
- なんで元と同じ計算だと言えるの？
  - Proof 9.1に証明があります
- そのCは本当に素敵なの？計算の形がちょっと変わっただけじゃないの？
  - その通りです、が……
  - この形にしかパラメータが復元できないのです

Theorem 1

- 以下のGが存在すれば、Cが存在する

1. *For all rules* $a \to b\ c$, $C^{a \to b\ c}(y) = G^c T^{a \to b\ c} \mathrm{diag}(y G^b S^{a \to b\ c}) Q^{a \to b\ c} (G^a)^{-1}$
2. *For all rules* $a \to x$, $c^{\infty}_{a \to x} = 1^\top Q^{a \to x} (G^a)^{-1}$
3. *For all* $a \in \mathcal{I}$, $c^1_a = G^a \pi^a$

※ G以外の大文字は隠れ状態を引数とする
パラメーター行列

テンソルで書きなおしたけど、これからどうする？

1. Observable representationを作る
   - ここでSVDが用いられる
- 2. Cがobservable representationから閉じた形で計算できることを示す

SVD前の準備：素性ベクトル

- inside tree, outside treeから素性ベクトルを作る
  - 詳しくは書いてないが、高次元疎ベクトルを想定している模様
  - φ(t) ∈ R^d (t: inside tree)
  - ψ(o) ∈ R^d' (o: outside tree)
- この辺りの工夫が提案手法の貢献
  - ツリーのままだとSVDで扱えない

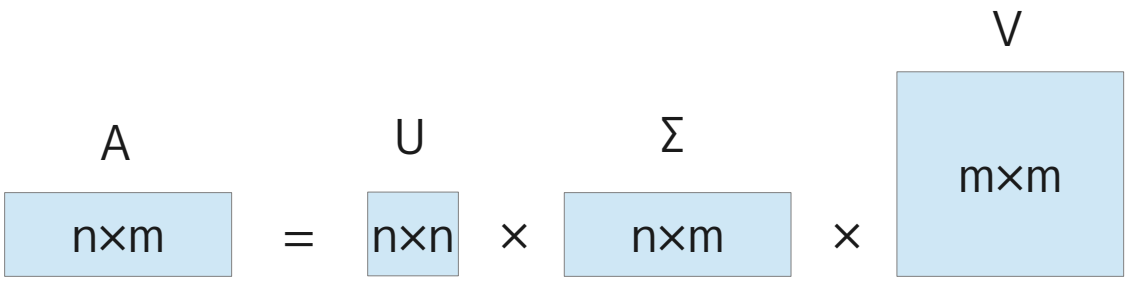
SVD用の行列を作る

以下を全てのルールに対して行う

- トレーニングデータ中からaを使っている箇所をM回サンプリングしてくる
- それぞれの箇所からφ, ψを計算し、平均を取る
- Ω = φ ψ ^Tというd'xd次元の行列を作る

SVDってなに？

- 特異値分解、Singular Value Decomposition
- n×m行列をn×n行列 × n×m行列 × m×m行列に分解する手法
- $A = U\Sigma V$ と分解したものとすると、
  U と V は直交行列（つまり $U^{\top}U$ は単位行列）
  Σ は対角線上に特異値が並び、それ以外の場所は 0



SVDする

- 使うのはThin SVD (特異値が大きい物から順にm個だけを取ってきてくるタイプのSVD)
  - m: 隠れ状態の数

Observable Representation用の確率変数を定義する

$$Y_1 = (U^{a_1})^\top \phi(T_1) \quad Z = (V^{a_1})^\top \psi(O)$$

$$Y_2 = (U^{a_2})^\top \phi(T_2) \quad Y_3 = (U^{a_3})^\top \phi(T_3)$$

ルール毎にSVDを行うので、UやVの方にはルールのIDが載っている。
φやψはベクトルを返す関数で、そこに行列を掛けているので、結局d次元のベクトルをm次元へと線形変換している。
Ω（φΨの期待値で作った行列）のSVDで得られた特異行列を使って線形変換するという事は、φやΨの情報量をできるだけ落とさないように作られた行列を使って変換しているということになる。

Observable Representation

- 隠れ状態を用いず計算できる統計量のこと

$$
\begin{aligned}
\Sigma^a &= \mathbf{E}[Y_1 Z^\top | A_1 = a] \\
D^{a \to b\, c} &= \mathbf{E}\left[[[R_1 = a \to b\ c]] Y_3 Z^\top Y_2^\top | A_1 = a\right] \\
d^\infty_{a \to x} &= \mathbf{E}\left[[[R_1 = a \to x]] Z^\top | A_1 = a\right]
\end{aligned}
$$

隠れ状態が出てこないので、サンプリングして
カウントして平均を取れば期待値は計算できる

Observable representationの利点

- 簡単に計算可能
- 隠れ状態を持つパラメーターと等価な情報がその中に含まれている
  - Observable representationを組み合わせると、パラメーターテンソルCが計算できる

Dは式変形で以下のような形に変形できる

$$D^{a\to bc}(y) = G^c T^{a\to bc}\mathrm{diag}(yG^b S^{a\to bc})Q^{a\to bc}\mathrm{diag}(\gamma^a)(K^a)^\top$$

$$cf.\, C^{a\to bc}(y) = G^c T^{a\to bc}\mathrm{diag}(yG^b S^{a\to bc})Q^{a\to bc}(G^a)^{-1}$$

パラメーターテンソルCの計算

$$
\begin{aligned}
C^{a\to b\ c}(y) &= D^{a\to b\ c}(y)(\Sigma^a)^{-1} \\
c^{\infty}_{a\to x} &= d^{\infty}_{a\to x}(\Sigma^a)^{-1} \\
c^{1}_{a} &= \mathbf{E}\left[[[A_1 = a]]Y_1 | B = 1\right]
\end{aligned}
$$

Spectral Leraningは なぜうまくいくのか？

- Dが隠れ状態を考慮せずに計算できるのに、隠れ状態を含んだ式としても書けるのが重要
- 問題をあらかじめテンソルCを使って定義しなおしたことも重要
  - 元のパラメーターを求めることはできない
    - できなくはちょっと言い過ぎだが、Hsuらはwe believe it to be generally unstableと述べている
    - AnandkumarらはCOLT2012で元のパラメーター行列を復元できるちょっと違った手法を提案している

性能はどれぐらいか？

- 計算時間はEMより圧倒的に速い
- 発表スライドによると、F1値はEMよりも1〜2%程度低いので、その改善が今後の課題とのこと

感想

- 隠れマルコフモデルは品詞の視点から見ると教師なしのクラスタリングとも捉えられる
- そう考えるとHsuらのアイデアは自然である
- 本質的にはSVDがこのマジックのような現象を生み出している
  - クラスタリングが局所解なしでできるから

まとめ

- Spectral Learningを中心に、面白かった発表をいくつか紹介した
- 来年のACLはブルガリアです