Phrase Patterns for Text Classification

speaker: Hirakata Kennai

Univ. of Tokyo, CS

October 27, 2014

読んだ論文

“Learning Phrase Patterns for Text Classification”
Author: Bin Zhang+
http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6457440

導入

テキスト分類のための素性として n-gram が通常使われる
ある程度の精度は達成されてる
ドメインに特化してしまい一般性がない
n-gram とそのまま置き換える素性として、phrase pattern がある

先行研究

- Wiebe+, 2005
  文章の subject を教師ナシで推定する．これは目的語を含んだフレーズパターンで分類した。
- Sun+, 2007
  第二外国語学習者の書いた誤文法を検出。
- Thur and Davidov, 2010
  Twitter や Amazon レビューから「皮肉」な文を検出
- Zhang+, 2010
  Speaker role

素 phrase pattern

文を語の列 $[w_1 \cdots w_n]$ とみなす. これに対して phrase pattern とは、語の列 $[u_1 \cdots u_m]$ と定める.
phrase pattern が 文にマッチするとは、subsequence の関係にあること

$$\forall i.\ u_i = w_{j_i}$$
$$i_1 < i_2 \implies j_{i_1} < j_{i_2}$$

phrase pattern with word classes

語の列でなく、word class も利用したい
word class としては、POS とか polarity とか (個数を制限しない)
word `w` の class として (文脈に依存して) {`c1 .. cn`} があるとき、

```
w -> {w, c1 .. cn}
```

という拡張を、文と phrase pattern に対して適用する.

文 $[w_1 \cdots w_n]$ (w_i はトークンとクラスの集合)
phrase pattern $[u_1 \cdots u_n]$ (同様に集合の列)
マッチすることの定義は以下のように

$$\forall i.\ u_i \subseteq w_{j_i}$$
$$i_1 < i_2 \implies j_{i_1} < j_{i_2}$$

パターンの学習

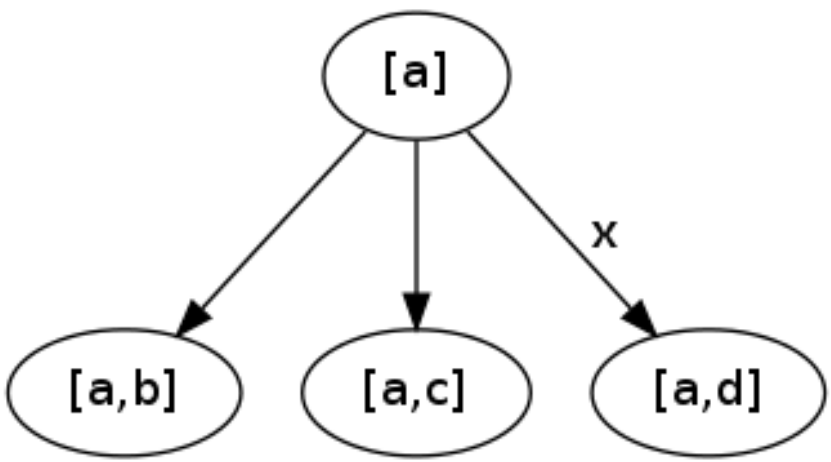
コーパス D から、意味のありそうなパターンの学習アルゴリズムとして

- PrefixSpan
- CloSpan

などがある．ここでは一つ目を紹介して、これを改良する．

PrefixSpan (Pei, Han+, 2001)

コーパス D (文の集合) から、頻度が閾値 f を上回るようなパターンを、頭から一つずつ word or class を追加していくことで得る
パターン $[a]$ が得られたら、それを伸ばすような $[a, X]$ も試すことで、パターンを学習していく.



コーパス D, 閾値 f に対して PrefixSpan$(D, \rho = [\,])$

Algorithm 1 PrefixSpan(D, ρ)

Require: D is a corpus and ρ is a prefix pattern

$P \leftarrow \emptyset$
for word or class a in D **do**
  $\rho' \leftarrow$ append(ρ, a)
  if matchFreq$(D, \rho') >= f$ **then**
    $P = P \cup \{\rho'\}$
    $D' \leftarrow \rho'$-project(D)
    $P' =$ PrefixSpan(D', ρ')
    $P = P \cup P'$
  end if
end for

ρ-project は、パターン ρ にマッチする文だけ抽出する射影

長さ n のパターンを見て、長さ $n+1$ のパターンを見て...
とやると計算効率が悪いので実際は、後ろに追加した1つだけ見ればよい。

Algorithm 2 ρ-project(D)

$D' \leftarrow \{\}$
$[w_1 \ldots w_m] = \rho$
for sentence $[b_1 \ldots b_n] \in D$ **do**
  if pattern ρ matches this sentence **then**
    find indecies j such that
    $w_1 = b_{j_1} \ldots w_m = b_{j_m}$
    $D' \leftarrow D' \cup \{[b_{j_m+1} \ldots b_m]\}$
  end if
end for

射影で、マッチより後ろ部分だけを抽出することで、
PrefixSpan で、パターンの頻度を確認するときに、新たに
追加した a だけを、見ればいい.

改良版 PrefixSpan

尺度として頻度を用いたが、分類器において、相互情報量が良い尺度になりうる.
あるパターンについて, マッチするかどうか $X = 0, 1$, 文書のクラス $Y = 0 \ldots K$

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

$$= \sum_{x,y} p(x|y) \log \frac{p(x|y)}{\sum_{y'} p(x|y')p(y')}$$

相互情報量の上限

パターン ρ がマッチするかどうか X, ρ を伸ばして得たパターン ρ' がマッチするかどうかを XE と書くと

$$p(XE = 1|y) \leq p(X = 1|y)$$

とあるから、次のような上限が存在する (前頁の I は $p(x|y)$ について上に凸!?)

$$\max I(XE; Y)$$

つまり、パターン ρ について、パターンを伸ばすことで増価しうる相互情報量の上限は予め算出できる

改良版 Prefix Span

尺度を 頻度 $\geq f$ から、相互情報量を使うように書き換える

Algorithm 3 ExtendedPrefixSpan(D, ρ)

$P \leftarrow \emptyset$
for a in D **do**
  $\rho' \leftarrow$append(ρ, a)
  if ρ' の相互情報量が閾値以上なら **then**
    $P \leftarrow P \cup \{\rho'\}$
  else if ρ' の相互情報量の上限が閾値以上なら **then**
    $D' \leftarrow \rho'$-project(D)
    $P' \leftarrow$ExtendedPrefixSpan(D', ρ')
    $P \leftarrow P \cup P'$
  end if
end for

実際に使う word class は以下の通り

- Lemma
- Word shape
- POS
- NE
- LIWC
- subjectivity lexicon
- manual
- automatic

Lemma

語の標準系を取り出す

{go, goes, going, went gone} → go

tool: NLTK WordNet lemmatizer

POS

tool: `Stanford log-linear POS tagger`
English models and trained models for Arabic, Chinese, French, Spanish, and German

Named entity (NE)

テキスト分類に於いてはこれが重要ということになっている

```
(sentence, word) ->
class ({Location, Person, Organization, Time, Date})
```

Stanford conditional random field-based NE recognizer (NER)
なるものが良いって。

LIWC dictionary

Linguistic Inquiry and Word Count (LIWC) は、単語を 64 の感情に関するクラスに分類する
Facebook が使ってた
文脈に依存せず、一つの単語について分析する。
`http://www.liwc.net/tryonline.php`
完全版は $89.95 で使える

MPQA subjectivity lexicon

MPQA さんが GNU GPL の元で配布してる辞書
単語とその品詞から引く形になっている

```
(word, POS) -> class
```

8222 項目が登録されてる

e.g.

```
type=weaksubj len=1 word1=dominate pos1=verb
stemmed1=y priorpolarity=negative
```

manual

そのトピックについて詳しい人間が手作業で、そのクラスに属する単語をリストアップしていく.
あとの実験で使われたものでは

```
AGREEMENT = [right, agree, true]
DISAGREEMENT = [doubt, inappropriate]
ALIGNMENT = AGREEMENT ++ DISAGREEMENT
MODAL = [could, should]
NEGATIVE_DISCOURSE_ORDER =
  [however, but, nevertheless]
```

automatic

Brown+, 1992 "Class-based n-gram models of natural language" の手法を用いる
1 次マルコフモデルを使った、word のクラスタリング
クラスタ数 = 10, 100, 1000

実験

n-gram (と他の素性) ではそれなりに難しいタスク

- Speaker role
- Alignment move
- Authority claim

Baseline を n-gram (3-gram までに制限) と他とするとき、
pattern (長さ 3 に制限) と他でやってみる
分類は 最大エントロピー法
5-fold cross validation で精度または F 値を出す

Speaker role

ニュースショー (音源) から、一つのセリフを発した人間の役割 (Host, Guest, Voice bite) を推定する
Liu+ 80%

data

- 48 English talks
- 90 Mandarin talks

の録音に対して、
REF (Reference human transcripts) と ASR (automatic speech recognition) output (using SRI Decipher ASR system) を対象にする.
ASR は、結構間違える. 英語については 22.8% 北京語については 38.6% くらい、単語/文字を誤る.
$\kappa = 0.67/0.78$

Result - English

pattern + word class	Ref.	ASR
n-gram (no pattern)	85.8%	85.0%
pattern w/out class	86.9	85.6
w/ lemma	86.8	85.4
w/ POS	86.2	85.8
w/ NE	86.9	84.7
w/ LIWC	86.0	85.9
w/ MPQA	86.5	85.9
w/ automatic	87.1*	85.6

Ref に対して、
ASR もそこまで
悪くない
n-gram もそんな
悪くないんだよね

Result - Chinese

pattern+word class	Ref.	ASR
n-gram	84.6	70.2
pattern w/ no class	85.8	77.8
w/ POS	84.8	74.5
w/ automatic	85.7	77.2

中国語はいくつかの素性が使えないからしょうがない。
あと POS が使い物になってないのが意外。

Alignment move

ネット上の議論においてある発言が趣旨に同意してるのか (positive) 反対してるのか (negative) を見る
neutral はない

data

実験で使うのは Wikipedia talk page

- 211 English pages and
- 225 Chinese pages

$\kappa = 0.50/0.53$
いくつかの文は pos, neg 両方を含む
あるアノテータが pos をつけて、あるアノテータが neg をつけたようなものを、両方あるものとして、pos+neg というラベルにする
分類は、pos/none, neg/none の 2 つの分類器を作って union をとる

Result - English (F-score)

pattern + word class	Ref.	ASR
n-gram (no pattern)	38.1%	38.8%
pattern w/out class	40.5	38.9
w/ lemma	40.2	38.8
w/ word shape	40.0	39.3
w/ POS	39.0	38.6
w/ NE	40.5	38.9
w/ LIWC	39.0	38.7
w/ MPQA	39.2	40.5
w/ manual	40.8	39.4
w/ automatic	40.7	40.5

Result - Chinese (F-score)

pattern + word class	Ref.	ASR
n-gram (no pattern)	26.7%	29.7%
pattern w/out class	31.2	31.2
w/ POS	32.7	30.7
w/ manual	33.9	31.5
w/ automatic	30.9	30.6

基本的に manual が強い

Authority claim

showing her knowledge or experience with respect to a topic, or using some external evidence to support herself

- forum claim: フォーラム内のソース (発言) を引用する
- external claim: 外のソースを引用する

引用を見るだけだから Unigram で実際けっこう良い
(Marin+, 2010)

data

- 339 English pages and
- 225 Chinese pages

発言ごとに、forum / external authority claim であるかどうか.
$\kappa = 0.59/0.73$
データは疎である. authority claim は全体の 20 % だった

Result - English (F-score)

pattern + word class	Ref.	ASR
n-gram (no pattern)	49.5%	46.5%
pattern w/out class	47.7	46.0
w/ lemma	48.0	46.7
w/ word shape	48.2	45.8
w/ POS	48.6	45.1
w/ NE	47.7	46.0
w/ LIWC	48.9	46.5
w/ MPQA	47.8	45.5
w/ manual	48.0	46.8
w/ automatic	48.9	46.1

Result - Chinese (F-score)

pattern + word class	Ref.	ASR
n-gram (no pattern)	32.2	32.3
pattern w/out class	31.8	33.5
w/ POS	34.3	40.3
w/ manual	30.3	37.9
w/ automatic	31.4	35.6

まとめ

- 基本的には
  n-gram → 素 phrase pattern → phrase pattern with word classes
  で強くなる
- word class は利用可能なら manual が強い
- Speech role の ASR で見たように、訓練データに頑強性がある

まとめ

- 基本的には
  n-gram → 素 phrase pattern → phrase pattern with word classes
  で強くなる
- word class は利用可能なら manual が強い
- Speech role の ASR で見たように、訓練データに頑強性がある
- この実験は長さ 3 に制限していたが本気を出したバージョンを見たかった