元テキストが復元不可能な部分文字列への ラベル付与によるテキスト分類

山下 達雄*，清水 伸幸（ヤフー株式会社）

{tayamash,nobushim}@yahoo-corp.jp

１．概要

テキストの分類タスクで学習データとして使用するための、ラベル付き正解コーパスをクラウドソーシングで作成する際に、コンプライアンス等の問題によりコーパス自体を外部に出せないケースがある。

本研究では、テキストを元のテキストが復元不可能なレベルの極大部分文字列に変換し、外部に出せるようにした上で、クラウドソーシングでラベル付けタスクを実施し、その結果をナイーブベイズの確率計算に直接利用するテキスト分類手法を提案する。

この手法により、テキストの一部分に対してのラベル付与のみで、一般的なテキスト分類タスクの精度に近い値を得ることができた。

２．正解ラベル付きコーパス

本研究で使用した正解ラベル付きコーパスは、Twitter の投稿をベースとして作成した。作成にあたっては、作業者が約 4 万件の投稿に対して感情ラベル Posi, Nega, Other の付与を行った。同じ投稿に対して作業者 2 名が作業を行い、両者の付与したラベルが同じもののみを採用した結果、全 22526 件となった。これを学習用コーパス 2 万件、テスト用コーパス 2526 件に分けた。正解ラベル付きコーパスのサンプルを表 1 に挙げておく。

ラベル	投稿テキスト
Posi	壇蜜さん綺麗(´ー`)?
Nega	録画なのに引っ張りすぎ…だから嫌われるんだよXXテレビ！
Other	ドデスカー押し kis-my-ft2 だって！

表 1: 正解ラベル付きコーパスの例

３．素性として使う極大部分文字列

本研究では、機械学習の素性とクラウドソーシングでのラベル付け単位として極大部分文字列を用いている[1]。

文字の代わり形態素を最小単位としており、正確には極大部分形態素列と呼ぶべきものだが、便宜上そのまま極大部分文字列と呼ぶ。

前述の 2 万件の学習コーパスから極大部分文字列を抽出すると 56,761 個となった。これらの極大部分文字列に表 2 のルールを適用しノイズを除去した結果、最終的に 14,140 個となった。

形態素解析を行い活用形の連続は一つにまとめる
前後のスペースは削除
2～10 文字以外のものは排除
前後が語の区切りにマッチしないもの（名詞の途中など）は排除
途中に文の区切りが入るもの（例「ん。ああ」）は排除
句読点や助詞などで開始・終了するものは排除
数字連続、顔文字切れ、カッコ片方のみ、URL 破片の排除
形態素解析を行い活用形の連続は一つにまとめる

表 2: ノイズ除去ルール

４．ベースライン評価

ベースラインとして、学習コーパスとテストコーパスを用いた一般的な機械学習による評価を行った。素性として学習コーパスから抽出された前述の極大部分文字列 14,140 個を用いた。機械学習ツールとして SVM (liblinear[2]) を使い、モデルを作成し、テストコーパスで評価した。結果を表 3 に示す。全体の精度(Classification Accuracy)は 0.8096 であった。

５．クラウドソーシングによるラベル付与

14,177 個の極大部分文字列対し、クラウドソーシングサービス「Yahoo!クラウドソーシング」[3]を使いラベル付けを行った。各極大部分文字列対し、クラウドソーシングのユーザ 3 名に Posi Nega Other のラベル付けを行ってもらった。ラベルの分布は Nega : Other : Posi がおよそ 1:6:3 の比となった。ラベル付け結果例を表 4 に示す。

	Nega	Other	Posi	ans
Nega	69	117	17	203
Other	64	1580	157	1801
Posi	7	119	396	522
sys	140	1816	570	2526

	Pre	Rec	F 値
Nega	0.4929	0.3399	0.4023
Other	0.8700	0.8773	0.8737
Posi	0.6947	0.7586	0.7253
Ave	0.6859	0.6586	0.6671

表 3: ベースライン評価

極大部分文字列	Posi	Other	Nega
立川駅	0	3	0
したくな	0	2	1
忘れません	2	1	0
RT した人	0	3	0
バカレア	0	3	0
クインテットも最終回	0	1	2
よろしくおねがい	2	1	0
5 月下旬	0	3	0

表 4: クラウドソーシングによる極大部分文字列へのラベル付与例

６．クラウドソーシング結果の Naïve Bayes への適用

前節で、クラウドソーシングで 3 名のユーザに各極大部分文字列に対して Posi, Other, Nega のラベルを振ってもらった。この各極大部分文字列（以降「語」と呼ぶ）のラベル分布を用いて Naive Bayes を行う[4]。あるカテゴリが与えられたときのツイート(doc)が生成される確率$P(\mathrm{doc}|C_x)$は語の出現確率の間に独立性を仮定すると$\prod_i P(w_i|C_x)$で計算できる（w_iは doc に含まれる語）。ここで、各クラスC_xごとの語wの出現確率$P(w|C_x)$に着目する。

$$P(w|C_x) = \frac{C(w, C_x)}{C(C_x)}$$

この条件付き確率を計算するためには、はじめに、特定の語wの各クラスC_xでの出現頻度$C(w, C_x)$が必要となる。クラウドソーシングの結果からは正確な値が分からないため、擬似的な値を計算することとする。計算方法は単純で、wの（クラウドソーシングに出す極大部分文字列を作った）コーパスでの出現頻度を得て、それをクラウドソーシングでのラベル分布にそって割り振るだけである。例えば、ある語w_1のコーパスでの出現頻度が 36、クラウドソーシング結果のラベル分布が Posi 2 名、Nega 0 名、Other 1 名とすると、36 を 2:0:1 で割り振った 24:0:12 が各クラスにおけるw_1の擬似頻度となる。

$$C(w_1, C_{posi}) = 24,$$
$$C(w_1, C_{nega}) = 0$$
$$C(w_1, C_{other}) = 12$$

次に、各クラスの頻度合計$C(C_x)$を求める。これは前述の方法で求めた全ての$C(w, C_x)$をC_xごとに集計すれば良い。

$$C(C_x) = \sum_w C(w, C_x)$$

これらにより擬似的な$P(w|C_x)$を求めることができ、Naïve Bayes のモデルでの推論が可能となる。

７．評価

前節で説明した方法で、クラウドソーシング結果である 14,140 個の極大部分文字列から計算された確率値を用いて、ベースラインと同じ設定で評価実験を行った。結果を表 5 に示す。全体の精度は 0.7522 であった。

	Nega	Other	Posi	ans
Nega	60	139	4	203
Other	64	1645	92	1801
Posi	27	300	195	522
sys	151	2084	291	2526

	Pre	Rec	F 値
Nega	0.3974	0.2956	0.3390
Other	0.7893	0.9134	0.8468
Posi	0.6701	0.3736	0.4797
Ave	0.6189	0.5275	0.5552

表 5: 提案手法の評価

８．おわりに

テキスト分類タスクにおいて、一般的な学習データであるテキスト＋ラベルではなく、そのテキストから抽出した極大部分文字列＋それに対してクラウドソーシングで付与されたラベルによる方法を提案した。文脈情報の欠如という不利な条件ながらもベースラインに近い結果が得られた。

本研究では単純に Naïve Bayes に適用しただけであり、例えばスムージング[4]等の工夫による精度向上が期待できる。また、クラウドソーシングでのラベル付与においてクオリティの問題があり、ウェブ検索等で前後の文脈を確認させる等のタスクの再検討も考えている。さらに、Naïve Bayes の前提である語の独立性がそもそもないため、モデルや素性の工夫も課題である。以上のように、本手法の精度向上の余地は大きく、引き続き研究を進めて行きたい。

参考文献

1. 岡野原大輔, 辻井潤一："全ての部分文字列を考慮した文書分類", 情報処理学会研究会報告 NL(187), September 2008.
2. Yahoo!クラウドソーシング, http://crowdsourcing.yahoo.co.jp/
3. Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin : "LIBLINEAR: A Library for Large Linear Classification", Journal of Machine Learning Research 9, pp.1871-1874, June 2008.
4. Christopher D. Manning, Prabhakar Raghavan and Hinrich Schutze : "Introduction to Information Retrieval", Cambridge University Press. 2008.