誤り文の自動生成による校正エンジンの学習

中島寛人　　山田剛

日本経済新聞社

{hiroto.nakajima, tk.yamada}@nex.nikkei.com

1. はじめに

文章の校正は、広範な業種・職種に必要とされている機能であり、特に新聞社を含む出版業にとっては極めて重要である。一口に校正と言っても様々なレイヤーがあるが、「文章に含まれる誤字・脱字などの誤りを検出する」という意味の校正は、例えば辞書との照合によって実現できる。しかし、このような意味の校正であっても、以下のような課題がある。

(1) 新語（特に固有名詞）の登録
(2) 企業・部門・ジャンルに固有の正誤ルールへの対応（例えば、日経新聞のマーケット面では「三菱自動車」を「三菱自」と表記）
(3) 「ら抜き言葉」「不適切用語」などの「誤りではないが、容認できない」表記の検出

このような校正を「文章（またはその一部）の正誤判定」と捉えると、Recurrent Neural Network (RNN)のような判別機によって判別する、という方法が考えられ、実際に日本語文章の RNN による校正には先行事例[1, 2]がある。ここで、判別機を学習させる際に課題となるのは、「正しい文を得ることは容易だが、誤った文を得ることは容易ではない」ということである。誤り文は記事等の作成・校正プロセスにしか存在しないため、収集できる事例は量も少なく、誤り方の網羅性も低い。

そこで、これらの課題を効率的に解決する方法として、本論文では「RNN による校正エンジン」を「校正済みの文章から擬似的に生成した誤り文」を利用して学習させる、というアプローチを提案する。擬似データによる学習には先行事例がある[3, 4]が、本論文ではこれを校正に適用する。このアプローチにより、学習データを任意の文書集合から自動的に大量に生成することができるため、例えば「先週ある部署で作成した文章を元にして、今日その部署で使う校正エンジンを生成する」というような実務的に柔軟な運用も可能となる。

2. RNN による学習モデル

本論文の提案モデルは、「固定長の文字列を一文字ずつ続けて入力し、その文字列に誤りが含まれているかどうかを判定する」という方法をとる。

例えば、入力するサンプルとして、本稿の第 1 章から連続する 10 文字を適当に抽出すると、

「とは容易だが、誤った」

という「正しい文字列」が得られる。これに対して、上記の文字列とよく似た

「とは容易だむ、誤った」

は、標準的な日本語の文章に現れることは恐らくなく、従って「誤った文字列」と見なすことができる。このような文字列を生成する方法については次章で説明する。

上記のような文字列を判別機(RNN)で正誤判定する具体的な手続きは以下の通りである。

(1) 文字を one-hot-vector に置き換える（次元数は文字種数と一致）
(2) one-hot-vector を FFNN により embedding し、次元削減する
(3) 2 層以上の RNN (LSTM)に embedding vector を一文字ずつ順に入力する
(4) 文字列をすべて入力した際の RNN の最終層の出力を単純な FFNN に入力し、その出力を Soft-max 関数にかけて二値判定を行う

ここで、上記(2)(3)(4)はすべて NN により実現されているため、逆伝播計算により同時に最適化される。すなわち、特に embedding について、他の目的に合わせた変換ではなく、この正誤判定に適した変換が得られる。

本モデルにおける典型的な NN のサイズは下記の通りである。これを変更した場合の性能変化については、後に議論する。

one-hot-vector	約 4000 次元
embedding	約 4000 次元→256 次元
LSTM	256 次元→128 次元→64 次元

3. 誤り文の自動生成

前述の通り、本論文で提案するアプローチでは、正しい文字列を元に誤った文字列を自動生成する。ここでいう「誤った文字列」とは、第 1 章で述べた「誤字・脱字などの誤りを含む文字列」を意味する。ここで、例として先に挙げた

「とは容易だむ、誤った」

は、本稿に実際に現れる文字列の一文字を適当に置換しただけであるが、誤った文字列のサンプルとなっている。すなわち、正しい文字列に「ノイズ」を加えると、得られる文字列は高い確率で誤った文字列となる。さらに、ノイズの加え方を工夫することにより、現実に起こりやすい誤りに近づけることができ、結果的に現実の誤りを捕捉する確率も高まると期待できる。具体的には、以下の五通りの方法でノイズを加える。

(1) 置換：文字列中の任意の一文字を、同じ文字種（ひらがな／カタカナ／漢字）の別の一文字に置換する
(2) 並び替え：文字列中の連続する同じ文字種の二文字を並び替える（並び替えの対象となる箇所が複数ある場合は、任意の一箇所）
(3) 追加：文字列中の任意の一文字の直前あるいは直後に、同じ文字種の任意の一文字を追加する
(4) 重複：文字列中の任意の一文字を重複させる
(5) 削除：文字列中の任意の一文字を削除する

それぞれの方法により生成される誤りの具体例は下表の通りである。

元の文	今日は最高のスキー日和（だ）
置換	今日る最高のスキー日和
並び替え	日今は最高のスキー日和
追加	今日は最人高のスキー日
重複	今日は最高のスキキー日
削除	今日は最高のスキー和だ

「元の文」の末尾にある「(だ)」は、削除を行った際に文字列の長さを保つための予備である。こうして生成した五個の文字列は、標準的な日本語の文章に現れることは恐らくなく、従って「誤った文字列」と見なすことができる。

上記の方法では、ノイズを加えた文字列が正しい文字列になる可能性はある。しかし、日本語に用いられる文字の種類の多さを鑑みると、その確率は十分に低く、結果として学習結果に重要な影響を与えないと期待できる。

4. データセット

判別機(RNN)にトレーニング用サンプルとして読み込ませる文字列は「校正済みの文章」から切り出す必要がある。本論文の実験では、文章セットとして「日経テレコン」の「日本経済新聞電子版セクション」に登録されている記事を採用した。

また、新聞記事はジャンルによって語句・表現の出現頻度が大きく異なることや、特定のジャンルの記事でのみ使われる略称・略語があること等を踏まえて、記事ジャンル別に判別機を学習させる。記事ジャンルとしては、下記リンク先の「中分類」を採用した。

http://t21help.nikkei.co.jp/reference/cat845/post-563.html

記事からサンプルを抽出する上では、まず指定した文字数の文字列を適当に切り出し、そこに英数字および記号（句読点を除く）が含まれていれば、サンプルから除外した。これは、

(1) 英数字や記号は出現頻度が低く、十分なサンプルを得られない
(2) 数字はノイズを加えたものが正しい文字列となる確率が高い
(3) 英字の誤り検出は他の手法でも比較的容易に対応できる
(4) 英数字や記号は自動校正のニーズが小さい

などの事由を考慮したためである。

サンプルの抽出元となる記事の本数は諸条件により変化するが、典型的には、長さ 10 文字のサンプル 10 万個を抽出するのに必要な記事の本数は 1000 記事である。実験に利用するサンプルを抽出できる最小限の本数（1000 本, 1500 本などの概数）の記事を直近のものから採用した。また、トレーニング用とテスト用のサンプルは記事の段階で別個のものとした。これは、判別機の性能が過大に評価されることを防ぐため、および、より実運用に近い実験環境とするためである。

5. 実験

5.1 評価方法および評価指標

学習済み判別機の性能評価には、トレーニング用サンプルと同様の方法で自動生成したテスト用サンプルを利用した。これは、「誤った文字列」のサンプルを大量に収集、あるいは人手で生成することは困難なためである。テスト用サンプルの数は常にトレーニング用（典型的には 10 万個）の 10 分の 1 とした。

また、評価指標は

・FP：正しい文字列が誤と判定された割合

・FN：誤った文字列が正と判定された割合

とした。誤り検出の性能は F 値で評価することが一般的であるが、記事の作成・校正過程では誤りは稀ではないことから、上記の指標を採用した。また、実用上の要請からは両者がともに低い数値となることが望ましいが、特に新聞記事の校正に焦点を当てると、後者がより重視される。

5.2 記事ジャンルによる比較

記事ジャンルによって、語彙の多様性や新しい語彙の出現頻度が異なることから、判別機の性能も異なると予想される。そこで、複数のジャンルを対象として実験を行なった。共通の条件は下表の通りである。

RNN の構造	256>128>64>32
サンプルの長さ	10 文字
サンプルの個数	トレーニング：10 万個
誤り文の生成	各抽出サンプルに ランダムに 2 通りを適用

実験の対象としたジャンル、およびそれぞれの性能は下表の通りである。

ジャンル	FP	FN
企業/事業組換	22.4%	11.6%
政治/政治運営	18.4%	14.4%
社会/社会問題	23.1%	15.7%
技術/技術	22.1%	13.1%
経済/マーケット	13.9%	9.9%

上表の通り「経済/マーケット」が最も良い性能を示した。これは、話題が上場企業の株価や業績に限定されており、新出語彙が少ないためと考えられる。そこで、さらに検索タグ「東京証券取引所」が付与されている記事に限定したところ、性能は FP 12.8%, FN 6.5%に向上した。この条件を満たす記事は約 5 本/日で、実用上の粒度の下限に近いと考えらえることから、以下ではこれを実験対象として検証を進める。

【実際の記事：「東京証券取引所」タグ】

上げ幅は 200 円を超え、一時 1 万 9766 円まで上昇した。国連安全保障理事会が日本時間 12 日朝に採択した北朝鮮への制裁決議で石油の全面禁輸を見送り、米朝対立が一段と緊迫化するとの警戒感が和らいだ。このため、投資家の運用リスク回避の姿勢が弱まり、12 日の東京株式市場では……（後略）（2017/9/12, 560 文字）

5.3 RNN の構造による比較

本論文の提案モデルは embedding の後に適当な段数・ノード数の LSTM を配置する。本節では、この構造と判別機の性能の関係を観察する。共通の条件は下表の通りである。

サンプルの長さ	10 文字
サンプルの個数	トレーニング：10 万個
誤り文の生成	各抽出サンプルに ランダムに 2 通りを適用

実験の対象とした RNN の構造、およびそれぞれの性能は下表の通りである。

RNN の構造	FP	FN
256>128>64	13.0%	7.1%
256>128>32	11.9%	7.5%
256>64>32	14.0%	6.8%
256>128>64>32	12.8%	6.5%
256>128>64>32>16	12.5%	6.7%
256>128>64>32>16>8	13.4%	6.1%

上表の通り、RNN の構造が性能に与える影響は小さい。しかし、サンプル数や文字数を増加させた際に、小さい RNN では十分に学習できない恐れがあることから、以下では 6 番目を採用する。

5.4 サンプル数および誤り文による比較

本論文で提案するアプローチでは、誤りの生成パターンは 5 通りあるが、その適用方法によって性能は変化すると考えられる。そこで、この条件やサンプル数を変えて実験を行なった。共通の条件は下表の通りである。

RNN の構造	256>128>64>32>16>8
サンプルの長さ	10 文字

サンプル数と誤り生成パターンの組み合わせ、およびそれぞれの性能は下表の通りである。

サンプル数	誤り文	FP	FN
10 万個	(A)	8.8%	11.5%
20 万個	(A)	7.8%	7.6%
40 万個	(A)	7.4%	6.2%
10 万個	(B)	13.4%	6.1%
20 万個	(B)	10.2%	5.0%
40 万個	(B)	8.1%	5.3%
10 万個	(C)	19.6%	3.0%
20 万個	(C)	15.1%	2.6%
40 万個	(C)	13.3%	2.7%

(A) 各サンプルにランダムに 1 通りを適用

(B) 各サンプルにランダムに 2 通りを適用

(C) 各サンプルに 5 通りを 1 回ずつ適用

上表の通り、
・サンプル数が増えると FP, FN 共に改善
・誤り文が増えると FP は悪化し FN は改善
という傾向が認められた。ただし、(C)はサンプルの偏りにより適切な学習ができていない可能性があるので、以下では(B)を採用する。

5.4 サンプル文字列の長さによる比較

文字列の長さについて、短過ぎれば正誤判定に十分な情報がなく、長過ぎればトレーニング用とテスト用の一致度は下がると想定される。そこで、文字列の長さを変えた場合の判別機の性能を観察する。共通の条件は下表の通りである。

RNN の構造	256>128>64>32>16>8
サンプルの個数	トレーニング：40 万個
誤り文の生成	各抽出サンプルに ランダムに 2 通りを適用

実験対象とする文字列の長さ、およびそれぞれの性能は下表の通りである。

文字列	FP	FN
7 文字	11.0%	5.6%
9 文字	9.5%	4.7%
11 文字	8.3%	4.9%
13 文字	8.6%	4.3%
15 文字	9.8%	3.8%
17 文字	9.8%	4.0%
19 文字	11.8%	3.4%

上表の通り、FP は 11,13 文字で最良となるのに対して、FN は文字列が長くなるほど改善する傾向が認められた。

6. 実験結果の考察

6.1 誤判別の傾向

誤判別の傾向を見るため、5.4 節で文字列の長さを 13 文字とした際の実験結果から、特に Softmax の値が低いサンプル（下表に例示）を観察した。

	サンプル文字列	Softmax
FP	。株価指数先物への売りが落	0.500009
FP	日の欧米株高やギリシャ債務	0.500042
FP	ーベラスの下げが目立つ。住	0.500174
FN	配検討すると伝わった楽天が	0.500169
FN	て上値重くなる場面もあった	0.500634
FN	角が上げに転じるなど相場志	0.500902

観察した限りにおいて、FP サンプルでは誤判別の原因は不明確だったが、FN サンプルは著者らにも誤りの箇所を特定できず、誤判別とは言えない場合もあった。そこで、Softmax 値が 0.9 超であるサンプルに限定したところ、テスト 12 万件のうち 110,092 件が該当し、FP 5.8% FN 2.3%であった。したがって、特に FN については、この数値の方がより正当な性能評価に近い可能性がある。

6.2 実践的な事例での評価

5 章では自動生成したテスト用サンプルにより性能評価を行なったが、実践的な事例での判定を観察するため、抽出した文字列の一部を対義語に置換した正解サンプルと、同音異義語に置換した誤りサンプルを人手で作成した。その例を下表に示す（置換した箇所を網掛け）。

対義語	た武田が上昇した。産業革新
対義語	し、目先の反落を期待した買
対義語	った時価総額の小さい主力株
同音異義語	加と信用売り残の造花でとも
同音異義語	準で推移している。円安貴重
同音異義語	上昇も交換された。前日のボ

作成したサンプルを 5.4 節の学習済み判別機にかけたところ、判定成功は 86/100 個と 86/100 個、すなわち FP 14.0%, FN 14.0%であった。これらの文字列がトレーニング用サンプルと一致する確率は極めて低く、その判定結果は判別機の汎化性能を示している。以上より、実践的な事例でも 5 章と大きく乖離しない性能を示すことを確認した。

【参考文献】

[1] 高橋諒（リクルートテクノロジーズ）LSTM と Residual Learning でも難しい「助詞の検出精度」を改善した探索アルゴリズムとは www.atmarkit.co.jp/ait/articles/1611/11/news016.html
[2] Yuta Hitomi, Hideaki Tamori, Naoaki Okazaki and Kentaro Inui. Proofread Sentence Generation as Multi-Task Learning with Edit Operation Prediction. In Proceedings of the 8th International Joint Conference on Natural Language Processing.
[3] 斉藤いつみ，鈴木潤，貞光九月，西田京介，齋藤邦子，松尾義博（NTT）擬似データの事前学習に基づく encoder-decoder 型日本語崩れ表記正規化 自然言語処理学会年次大会講演集, 2017.
[4] 澤井裕一郎，進藤裕之，松本裕治（NAIST）文法誤り訂正のための疑似誤り生成によるラベルなしコーパスの利用 自然言語処理学会年次大会講演集, 2017.