Vol.2015-NL-220 No.4
2015/1/19

テキストストリームからの新エンティティの即時的検出

槇 佑馬[1]　吉永 直樹　鍛治 伸裕　喜連川 優

概要：Twitter などのテキストストリームには現実世界で次々と生まれる新しいエンティティに関する情報が発信される．そのような膨大な情報を理解するには，エンティティ単位で情報を整理することが有効であるが，そのためには何がエンティティであるかを把握しておくこと，すなわち知識ベースに載っていない新エンティティをできるだけ早期に検出して知識ベースに登録することが重要になる．提案手法では，テキストストリームから未知のエンティティ文字列を検出する手法を，テキストストリームから新エンティティ候補文字列を抽出し，候補文字列に対して新エンティティかどうかを識別するという手順で解くことを提案する．前者のタスクは形態素 n-gram と文字列の出現頻度を手がかりに行い，後者のタスクに関しては品詞などの言語情報を用いて教師有り学習で解く．エンティティ単位で評価を行った結果，適合率が 67.2%，再現率が 77.6%という結果になった．

1. はじめに

近年，モバイル端末の普及などによって多くの人が盛んに Web 上にテキスト情報を投稿するようになっている．例えば，Twitter や Facebook あるいはブログには，現実世界の様々な出来事やものや人に関する情報が投稿されている．また，Web 上で読めるニュース記事からも現実世界の出来事を知ることができる．これらのメディアはテキストの中でも特に時系列の順番で次から次へと投稿されることから，テキストストリームと呼ばれている．テキストストリーム上には，現実世界の現在の情勢に関する情報や，ある製品に対する意見や評判など有益な情報が載っているため，企業の戦略決定や災害時の情報抽出などのための重要な情報源となっている [1], [2].

実世界テキストストリームの内容を理解し，有益な情報をリアルタイムで抽出するためにはエンティティを認識すること重要である．あるエンティティを適切に認識するためには，知識ベース上にそのエンティティに関する情報を持っている必要があるが，あらゆるエンティティの情報を含んだ知識ベースを構築することは不可能である．というのも現実世界では新しい映画や組織が次々と誕生し続け網羅すべきエンティティが際限なく増えてしまうからである．また，これらの新エンティティ全てを人手で即座に知識ベースに登録していくことは，現実的ではない．

エンティティを認識する固有表現抽出 [3], [4] やエンティティリンキング [5], [6], [7], [8], [9], [10] の研究の多くは，まとまったテキストの中から言語的な特徴を用いてエンティティを一気に獲得する方法が取られているため，テキストが時系列に沿って小出しに流れてくるテキストストリームに適した手法にはなっていないことが問題となる．

本研究では，現実世界の情報が良く反映される Twitter などのテキストストリーム上に現れる新エンティティをいち早く検知し，知識ベースを自動で更新することを目的として，新エンティティと考えられるのある文字列をテキストストリームから検知する手法の検討を行った．実験では，Twitter のテキストストリーム上に出現する新エンティティとそうでない文字列を与え，文中での言語情報と，過去の出現に基づく言語情報の二種類を用いて，新エンティティであるかそうでないかの識別を行った．その結果エンティティ単位で適合率 67.2%，再現率 77.6%という結果が得られた．

本稿の構成は以下の通りである．まず 2 章で本研究で取り組むタスクについて説明する．次に，3 章で提案手法について述べ，4 章でその評価を行う．5 章で関連研究について述べる．そして，6 章でまとめと今後の課題について述べる．

2. タスク設定

本章では，本研究で取り組むテキストストリームからの新エンティティの検出について説明する．

エンティティの検出にあたって，何をエンティティとみな

[1] 東京大学
University of Tokyo
[†1] 現在，情報通信研究機構
Presently with NICT
[†2] 現在，国立情報学研究所
Presently with NII

表 1　テキストストリームの例
Table 1　example of text stream

12/15 13:05:02	第 3 次安倍内閣へ向け閣僚は全員再任
12/15 12:59:50	安倍晋三首相の続投が決まりました．
12/15 12:58:25	安倍氏が祀られている晴明神社へ行ってきた!

すのかということが問題となるが，本研究では既存研究 [11] に倣って知識ベースとしてよく用いられる Wikipedia 日本語版 [*1] の項目をエンティティと定義する．従って，この研究で検出の対象とする新エンティティは，まだ Wikipedia の項目が作られていないが，将来作られる可能性がある語と定義できる．これは，本研究の目的が，人手で整備するよりも早く自動で新エンティティを検知することであるため，Wikipedia に人手で登録された時点で目的を達成できなくなるためである．

次に，本研究でエンティティ検出の対象となるテキストストリームについて説明する．従来の自然言語処理タスクでは，解析対象がまとまった形で一度に与えられることを前提としているが，実際のテキストは時系列で連続的に与えられる．このような入力をテキストストリームと呼ぶ．入力文書が一度に与えられた場合，例えば同一文書に出現する”安倍”という文字列が指す対象は同一である場合が多く，こうした文脈情報を手がかりにして言語処理を行うことができる．しかし，テキストストリームでは，ある文と，その直後に出現する文に出現する”安倍”という文字列が指す対象は必ずしも同一ではない．また，テキストストリームでは，時系列の情報が明示的に与えられるという性質もある．ここで制約として，テキストストリームでは全てのテキストを読み込んでから処理を行うことができないとする．

本研究では，テキストストリームから新エンティティを検知するために，次のような設定を考える．まず入力として，例えば**表 1** のような時系列テキストの列が時系列の順番に与えられる．この中から”第 3 次安倍内閣”（Wikipedia に項目がないとする）を新エンティティと判断することが正解となる．”安倍晋三”（Wikipedia に載っているエンティティ），”第 3 次安倍内閣へ”や”続投”（エンティティではない文字列）を新エンティティとして検知した場合不正解となる．

3. 提案手法

本章では，テキストストリームから新エンティティを検知するための手法を提案する．
以下の手順で新エンティティを検出するまず，テキストストリームから新エンティティになりうる文字列を抽出する．提案手法では形態素 N-gram の情報と出現頻度の情報を用いて，新エンティティ候補文字列を抽出する．次に，

[*1] http://ja.wikipedia.org/

表 2　Wikipedia 項目の形態素 n-gram 分布
Table 2　morpheme n-gram distribution on the set of wikipedia titles

形態素 n-gram	項目数	割合（%）	累積（%）
1-gram	399449	26.8	26.8
2-gram	435485	29.2	56.0
3-gram	324496	21.8	77.8
4-gram	136334	9.2	86.9
5-gram	89872	6.0	93.0
6-gram	39582	2.7	95.6
7-gram	23307	1.6	97.1
9-gram	19777	1.3	98.5
9-gram	9025	0.6	99.1
10-gram 以上	13064	0.9	100
all	1490441		100

抽出した新エンティティ候補が新エンティティであるかないかを識別する．提案手法では新エンティティ候補文字列が出現した文の言語情報や，過去の出現に基づく情報を素性とし，SVM を学習して識別問題を解く．

3.1 新エンティティ候補文字列の抽出

このステップでは，テキストストリームから新エンティティになりえる候補文字列を探すことが目的となる．このステップでは，テキストストリーム中の各文から，新エンティティの候補となる文字列を列挙する．単純に，全ての部分文字列を新エンティティ候補として列挙した場合，次ステップで新エンティティ候補が実際に新エンティティかどうか判定する分類問題が難しくなる可能性がある．また，分類コストの増加から処理のリアルタイム性が担保で着なくなる可能性がある．これらを考慮すると，このステップでは，可能な限りエンティティとなりえない文字列を排除しつつ，かつエンティティを漏らさないように候補を列挙することが目標となる．そこで我々は，テキストストリームの各文に対して形態素解析を行い，形態素 9-gram 以下の全ての文字列のうち，一定期間内に 10 回以上出現した文字列のみを新エンティティ候補文字列とした．候補を形態素 9-gram 以下に制限したのは，知識ベースである Wikipedia に登録されたタイトルタイトルの約 99 %が形態素 9-gram 以下であったためである（**表 2**）．また，出現頻度の極端に少ない文字列は扱う意義が薄く，また新エンティティかどうかを判定するのに必要となる統計量も十分に集まらないため，候補文字列を効率よく削減する目的で制限を設けた．

3.2 新エンティティ候補文字列の識別

このステップでは，前節で選択した新エンティティ候補文字列について新エンティティであるかそうでないかを識別する分類器を学習する．その際，分類器としてはサポー

表 3　SVM の学習に用いる素性
Table 3　features for training SVM

素性の種類	対象文字列	素性
静的な素性	候補文字列	形態素長
		形態素
		語頭の形態素
		語末の形態素
		品詞
		品詞（細分類）
		新エンティティとの類似度
動的な素性	候補文字列	文頭, 文末に位置する
	直前直後の文字列	形態素
		品詞
		品詞（細分類）
累積の素性	過去に出現した際の動的な素性	

トベクタマシン（SVM）を用い，素性として新エンティティ候補文字列とその文脈を用いる，候補文字列が過去に観測された際の文脈を用いる
．素性は大きく 3 つのグループに分けられる．1 つ目は，新エンティティ候補の文字列から直接生成される，出現文脈に依存しない静的な素性．2 つ目は，新エンティティ候補の出現する文によって変わる動的な素性．3 つ目は，テキストストリームにおいて過去に新エンティティ候補が出現した際の動的な素性を累積した素性とする．具体的には**表 3** に示したものを素性として採用する．新エンティティとの文字列類似度として，Wikipedia の全項目名との最小編集距離を使用した．

4. 評価実験

本節では，提案手法により，新エンティティの抽出ができるかを確認するための実験結果について報告する..

4.1 実験設定

テキストストリームとして，喜連川豊田研究室で収集されている 2014 年 1 月 1 日から 2014 年 9 月 23 日までの Tweet データを用いた．このうち，Twitter の機能のうち，公式 RT されたものはデータから除いた．

学習と評価で利用するための正例は以下のように作成した．2014 年 09 月 24 日から 2014 年 09 月 30 日の間に Wikipedia に新規追加された項目のうち，リダイレクトや曖昧性のある語を除き，実験で用いるテキストストリームの中で 1 回以上出現した 295 項目を新エンティティとした．こうして定めた各新エンティティと表層が一致する形態素列を含む Tweet を集め，この結果，66730 個の正例（1 エンティティ候補あたり平均 226 個）を作成することができた．

次に，負例の作成は以下のように行った．まず，既知エンティティを負例に含めるため，Wikipedia に 2013 年 12 月 31 日以前に追加された項目への，リダイレクトをランダムに 292 項目抽出し，正例を作成したときと同様の手続きで負例を作成した．次に，エンティティではないランダムな文字列を負例に含めるため，2014 年 9 月 24 日から 2014 年 9 月 30 日の全ての Tweet の中から形態素 9-gram 以下で出現頻度が 10 回以上の文字列をランダムに取り出し，正例を作成した方法と同様にして 300 項目の新エンティティではない文字列を含む負例を作成した．上記の 2 種類の負例の集合をあわせて，144120 個の負例（1 エンティティ候補あたり 243 個）を作成することができた．

こうして作成した正例と負例から，それぞれ 237 と 478 のエンティティ候補のデータを抽出し，それらを学習用データとした．そして，残りを評価用データとした．

形態素解析には，Mecab[*2] と ipadic を，SVM の実装には，liblinear[*3] を利用し，カーネルは線形関数を用いた．

4.2 実験結果

実験の結果に対して 2 通りの評価を行う．1 つは，Tweet ごとに推定したラベルが正解ラベルと一致していれば正解とするトークンベースの評価，もう 1 つは，正例については Tweet セットを通して 1 度でも正例と判断されたら正解，負例については 1 度でも正例と判断されたら不正解とするタイプベースの評価である．システムによって新エンティティと判断されたエンティティは知識ベースに登録することを想定しているため，2 つ目のようなタイプベースの評価を行った．Tweet ごとに推定されたトークンベースの評価の結果，ラベル正解率は 72.9%であった．文字列種類ごとのタイプベースの評価の結果，適合率で 67.2%，再現率で 77.6%という結果となった．

4.3 適合率-再現率曲線

提案手法では，一度でも新エンティティと判定された文字列は知識ベースに登録される．そのため，誤って正例と判断する頻度が多いと，知識ベースの信頼性が落ちてしまうという問題がある．質の高い知識ベースを維持するためには，適合率が高くなるような識別をするのが好ましい．そこで，適合率と再現率のバランスを決定するパラメータを設定したときの適合率-再現率曲線が**図 1** である．

5. 関連研究

本節では，本研究と関連する研究について述べる．従来，エンティティリンキングに関する研究が数多く存在するが [5], [6], [7], [9], [10]，知識ベースに登録されているエンティティのみがリンキング対象となっており，新エンティ

[*2] http://mecab.googlecode.com/svm/trunk/mecab doc/index.html
[*3] http://www.csie.ntu.edu.tw/ cjlin/liblinear/

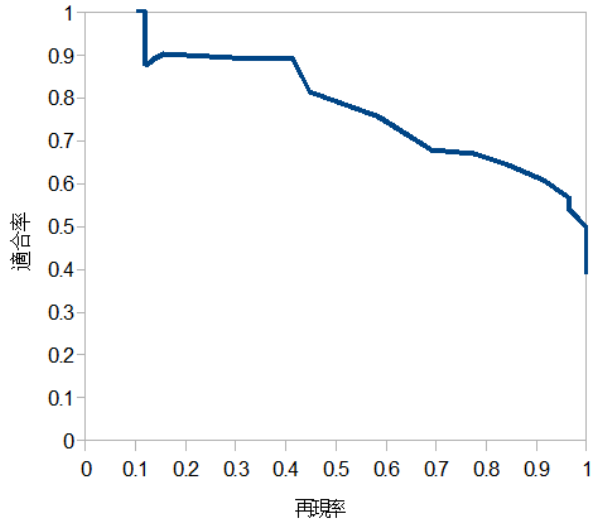

図 1 適合率-再現率曲線
Fig. 1 precision-recall curve

ティを扱った研究は数が少ない．Lin[3] らは，新エンティティの獲得に関する研究を行っているが，オンライン学習を行う方法については議論されていない．

6. おわりに

本稿では，実世界テキストストリームに対してエンティティリンキングを行う際に問題となる，知識ベースに登録されていない新エンティティをその場で検出するタスクを提案し，具体的にこれを教師有り学習を用いて解く手法を提案した．提案手法では，テキストストリームの各文から，エンティティとなりうる文字列を候補として列挙し，これを分類器を用いて新エンティティかどうか識別することで新エンティティの検出を行う．分類器の素性としては，候補文字列から直接生成される出現文脈に依存しない素性のほか，候補の前後の文字列などの出現文脈，さらにはストリーム中で過去に候補文字列が観測された際の出現文脈の履歴を素性として利用した．

実験では，Wikipedia に未登録のエンティティ（新エンティティ），登録済みのエンティティ（既知エンティティ），エンティティではない文字列に対するテキストストリームを用意し，このテキストストリームから未登録の新エンティティを識別する評価データを作成し，提案手法の有効性を評価した．

今後の課題としては，以下の 2 点が挙げられる

評価用データセットの整備 今回の実験では，Wikipedia に未登録のエンティティ，登録済みのエンティティ，エンティティでない文字列から，未登録の新エンティティを識別する評価データを作成し，提案手法の有効性を評価したが，実際にテキストストリームから新エンティティ候補を検出する際には false positive がどの程度あるか，が実用上大きな問題となる．この評価を行うため，人手で新エンティティをタグ付けデータセットを作成する予定である．

バーストの利用 新エンティティを認識することが特に重要となるのは，テキストストリーム中でエンティティに関する記述が（単位時間当たりで）増えた場合，すなわち，バーストが発生したときである．このようなバーストを分類器の特徴量や，候補列挙の際の条件として考慮することで，認識する重要性が高いエンティティをもらさず，かつ正確に検出することを目指したい．

参考文献

[1] Yue, Sui, and Yang Xuecheng. "The potential marketing power of microblog." Communication Systems, Networks and Applications (ICCSNA), 2010 Second International Conference on. Vol. 1. IEEE, 2010.
[2] Vieweg, Sarah, et al. "Microblogging during two natural hazards events: what twitter may contribute to situational awareness." Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2010.
[3] Lin, Thomas, and Oren Etzioni. "No noun phrase left behind: detecting and typing unlinkable entities." Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Association for Computational Linguistics, 2012.
[4] Nakashole, Ndapandula, Tomasz Tylenda, and Gerhard Weikum. "Fine-grained semantic typing of emerging entities." ACL, to appear (2013).
[5] Cucerzan, Silviu. "Large-Scale Named Entity Disambiguation Based on Wikipedia Data." EMNLP-CoNLL. Vol. 7. 2007.
[6] Ratinov, Lev-Arie, et al. "Local and Global Algorithms for Disambiguation to Wikipedia." ACL. Vol. 11. 2011.
[7] Bunescu, Razvan C., and Marius Pasca. "Using Encyclopedic Knowledge for Named entity Disambiguation." EACL. Vol. 6. 2006.
[8] Mihalcea, Rada, and Andras Csomai. "Wikify!: linking documents to encyclopedic knowledge." Proceedings of the sixteenth ACM conference on Conference on information and knowledge management. ACM, 2007.
[9] Liu, Xiaohua, et al. "Entity linking for tweets." Proceedings of the 51th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics. 2013.
[10] Guo, Yuhang, et al. "Microblog Entity Linking by Leveraging Extra Posts." Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. 2013.
[11] Lin, Thomas, and Oren Etzioni. "No noun phrase left behind: detecting and typing unlinkable entities." Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Association for Computational Linguistics, 2012.
[12] Hoffart, Johannes, Yasemin Altun, and Gerhard Weikum. "Discovering emerging entities with ambiguous names." Proceedings of the 23rd international conference

on World wide web. International World Wide Web Conferences Steering Committee, 2014.
[13] 浅原正幸, and 松本裕治. ”日本語固有表現抽出におけるわかち書き問題の解決 (自然言語).” 情報処理学会論文誌 45.5 (2004): 1442-1450.