情報処理学会論文誌 Vol.54 No.4 1641–1654 (Apr. 2013)

統計的機械学習を用いた歴史的資料への濁点付与の自動化

岡 照晃[1,a)] 小町 守[1,b)] 小木曽 智信[1,2,c)] 松本 裕治[1,d)]

受付日 2012年6月13日, 採録日 2013年1月11日

概要：生の歴史的資料の中には，濁点が期待されるのに濁点の付いていない，濁点無表記の文字が多く含まれている．濁点無表記文字は可読性・検索性を下げるため，歴史コーパス整備の際には濁点付与が行われる．しかし，濁点付与は専門家にしか行えないため，作業人員の確保が大きな課題となっている．また，作業対象が膨大であるため，作業を完了するまでにも時間がかかる．そこで本論文では，濁点付与の自動化について述べる．我々は濁点付与を文字単位のクラス分類問題として定式化した．提案手法は分類を周辺文字列の情報のみで行うため，分類器の学習には形態素解析済みコーパスを必要としない．大規模な近代語のコーパスを学習に使用し，近代の雑誌「国民之友」に適合率 96%，再現率 98%の濁点付与を達成した．

キーワード：自然言語処理，機械学習，歴史的資料，濁点，近代文語論説文

A Statistical Machine Learning Approach to Automatic Labeling of Voiced Consonants for Historical Texts

Teruaki Oka[1,a)] Mamoru Komachi[1,b)] Toshinobu Ogiso[1,2,c)]
Yuji Matsumoto[1,d)]

Received: June 13, 2012, Accepted: January 11, 2013

Abstract: Raw historical texts often include mark-lacking characters, which lack compulsory voiced consonant mark. Since mark-lacking characters degrade readability and retrievability, voiced consonant marks are annotated when creating historical corpus. However, since only experts can perform the labeling procedure for historical texts, getting annotators is a large challenge. Also, it is time-consuming to conduct annotation for large-scale historical texts. In this paper, we propose an approach to automatic labeling of voiced consonant marks for mark-lacking characters. We formulate the task into a character-based classification problem. Since our method uses as its feature set only surface information about the surrounding characters, we do not require corpus annotated with word boundaries and POS-tags for training. We exploited large data sets and achieved 96% precision and 98% recall on a near-modern Japanese magazine, Kokumin-no-Tomo.

Keywords: natural language processing, machine learning, historical texts, voiced consonant mark, near-modern literary style of Japanese

1. はじめに

日本語で初めての大規模均衡コーパスとなる**現代日本語書き言葉均衡コーパス**（**BCCWJ**）[3] が昨年公開された．これにより今，コーパスを利用した日本語研究が急速に増えつつある．

しかし，平安時代や明治時代といった古い時代の資料（**歴史的資料**）のコーパスは，現代語のコーパスほど整備が進んでいない．そのため日本語研究の大きな位置を占める歴史的研究に，コーパスを用いることはいまだ難しい．この整備が進まない原因の 1 つとして，**歴史コーパス**の整備に必要な**校訂**（**表記整理**）の作業コストが高いことがあげられる．

表記整理とは，歴史的資料の記述中にある**正書法に一致**

1 奈良先端科学技術大学院大学
Nara Institute of Science and Technology, Ikoma, Nara 630–0192, Japan
2 国立国語研究所
National Institute for Japanese Language and Linguistics, Tachikawa, Tokyo 190–8561, Japan
a) teruaki-o@is.naist.jp
b) komachi@is.naist.jp
c) togiso@ninjal.ac.jp
d) matsu@is.naist.jp

a) 表記整理前

ひめ君いかにつれ／＼におほえ給ふらん日ころになりぬれはれいのくむしやし給ふらんと心くるしうおほしやるかのしるしのあふきは桜のみへかさねにてこきかたにかすめる月をかきてみつにうつしたる心はへなとめなれたる事なれとゆへなつかしくもてなしたり

b) 表記整理後

「ひめ君、いかに、つれづれにおぼえ給ふらん、日ごろになりぬれば、れいのくんじやし給ふらん」と、心くるしうおぼしやる。かの、しるしのあふぎは、桜のみへがさねにて、こきかたに、かすめる月をかきて、みづにうつしたる心ばへなど、めなれたる事なれど、ゆゑ、なつかしく、もてなしたり。

図 1 表記整理前のテキストは米国議会図書館蔵『源氏物語』翻刻本文[*1]より抜粋した（花宴，丁数：5 オ，行番号：5–8）．濁点無表記の文字に濁点を付与し，歴史的仮名遣と一致しない仮名遣いはすべて歴史的仮名遣に修正し，反復記号である踊字の展開も行った．句読点・カギ括弧の挿入は岩波書店の**日本古典文学大系** 14 巻**源氏物語 1** を参考に実施した

Fig. 1 The text before textual criticism is an excerpt from **Transcription of the Tale of Genji manuscript book at the Library of Congress**[*1] (Hana-no-En, page: 5 オ, lines: 5–8). We labeled voiced consonant marks for mark-lacking characters, normalized Kana-usage to Historical Kana-usage, and expanded Odoriji. We inserted punctuation marks and brackets following **Nihon Koten Bungaku Taikei**, Vol.14, **Genji Monogatari 1**.

しない表記をコーパスユーザにとって扱いやすい形に修正する作業のことである．表記整理は歴史コーパスの可読性・検索性を上げるために実施され，現在はすべて人手で行われている．具体例として，**図 1** に表記整理を行う前 a) と行った後 b) のテキストを並べて示した．見比べると，明らかに表記整理を行った後の方が読みやすくなっているのが分かる．

現代の正書法に一致しない表記として，たとえば，図 1 a) の中には**濁点無表記文字**（mark-lacking character）が多く含まれている．これは「おほしやる（オボシヤル）」の「ほ（ボ）」のように，**濁点が付いていることが期待されるのに，濁点の付いていない文字**のことである．**表 1** を見ると，明治初期の資料でも濁音の仮名文字の約 83%（368/446）（総文字数の約 4%（368/8,423））が濁点無表記で書かれている．濁点無表記の文字をそのまま残しておくと，歴史コーパスのユーザにとっては読みにくく，検索にも不便である．そのため，濁点無表記文字を濁点文字（marked character）に置き換えるという表記整理作業が必要になる．この作業を**濁点付与**という．

ただし，表記整理を人手で行うのは非常にコストが高い．特に作業者が専門家に限られてしまうことが問題である．表記整理を行うためには，まず対象となる資料を読んで理解しなくてはならない．しかし，歴史的資料の中では現代とは語彙が異なるうえ，表記や語法の面でも多様であり，読解には専門的な知識が必要である．作業の信頼性を保証するためにも，作業者は歴史的資料の専門家に限定される．だが，そういった専門家を集めることは難しく，作業人員の確保が大きな課題となっている．

表記整理の対象となる資料の量が膨大であることも問題である．たとえば，国立国語研究所が構築した近代語のコーパス**太陽コーパス**[11] は総文字数約 1,450 万文字の規模[*2]である．これに対し，熟練した作業者でも 23 ページ分の資料（約 1 万 6,000 文字）の濁点付与に 1 日がかりで取り組む必要がある[*3]．上述のとおり，作業者を大量に確保することは困難であり，人海戦術をとることができない．そのため，大量の資料を少数人で少しずつ整備していくしかなく，整備を終えるまでに多大な時間が必要となっている．

また表記整理作業を人手で行う場合，熟練した作業者であっても単純なミスをおかすことがよくある．特に濁点付与の作業では，濁点を付与すべき文字を見逃してしまうことが多い．実際，2 人の作業者がそれぞれに行った濁点付与の結果を比較してみたところ，濁点を付けた個所の一致率は 84%であり，一致しなかった原因のほとんどが濁点の付け逃しによるものであった．また，コーパス整備の初期段階で 1 人の作業者が実施した濁点付与の結果を完成後のコーパスと見比べた．すると作業者が単独で濁点付与を行った結果では，濁点無表記文字のおよそ 7%を見逃していたことが分かった．

国立国語研究所では，さらなる近代語コーパスの整備が進められている．しかし，表記整理作業の負担から，現場ではその自動化を望む声があがっている．

そこで本研究では，統計的機械学習の手法を用いて，歴史的資料の表記整理を支援する技術の開発を目的とする．これは従来行われてこなかった新しい試みである．歴史的資料の表記整理はこれまですべて人手で行われてきたが，作業者不足と作業対象の膨大さから実施に高いコストを必要とした．また人手の作業にミスはつきものである．しかしその作業を計算機に行わせることで，大規模な表記整理も低コストで実現でき，単純なミスもなくすことができると考えられる．

機械学習を用いた表記整理支援の第 1 段階として，濁点付与の作業を取り上げ，識別学習を用いた濁点の自動付与に取り組んだ．最初の目標として濁点付与を取り扱ったのは，濁点付与のタスクが「濁点が抜け落ちた文字に濁点を補う」と明確で取り組みやすかったためである．またタスクが単純な割に，表記整理の中での必要性は高い．これはコーパスの可読性や検索性向上の観点から，濁点の抜け落ちている問題が無視できないためである．

[*1] http://www.ninjal.ac.jp/LCgenji

[*2] 2005 年 11 月段階のデータ

[*3] 1 日の総作業時間を 5〜6 時間とした場合

表 1 濁点と濁音の統計．**明六雑誌**[*4]第 1 号（2011 年 12 月段階のデータ，総文字数：8,423）中に含まれる濁点文字と濁点をつけることが可能な文字（mark-less character）の中から，実際の発音が清音の文字数と濁音の文字数の内訳を調査した

Table 1 The contingency table of observed frequencies of characters and voiceness. We counted number of voiced characters and voiceless characters among marked characters and mark-less characters in **Meiroku Zasshi** [*4], Vol.1 (Data of Dec. 2011. Number of characters: 8,423).

表記＼発音	濁音（ガ，ザ，ダ，…）	清音（カ，サ，タ，…）	計
濁点文字（が，ざ，だ，…）	78	0	78
濁点をつけることが可能な文字（か，さ，た，…）	368 （濁点無表記文字）	1,273	1,641
計	446	1,273	

濁点付与の対象となる未校訂資料は基本的に翻刻直後の生テキストであり，形態素解析しても濁点無表記のような多様な表記状態のため高い精度が期待できない．ゆえに濁点付与を実施する際，事前に単語の情報を取得して用いることは困難である．また，既存の歴史コーパスで単語の情報までアノテーションされたものもごくわずかしかないため，単語情報の付いた訓練用リソースの確保もままならない．そこで提案手法では，単語の情報をいっさい使わない文字ベースの処理を採用した．具体的には，濁点付与を文字ごとのクラス分類問題として定式化し，分類の素性にも分類対象文字の周囲の文字列の情報のみを使う．周囲の単語境界の情報や品詞の情報はいっさい使用しない．ゆえに，分類器の学習にも形態素解析済みコーパスを必要とせず，生の歴史的資料に対しても濁点付与を行うことができる．また提案手法では，各文字に対する分類をそれぞれ独立に行う方法を採用した．そのため，未校訂の資料からでも濁点文字からならば，事例を抽出して学習に使うことが可能である．この方法により，学習に使える校訂済みの資料が少ない中世や近世の資料の濁点付与にも提案手法を適用することができる．

2. 濁点自動付与に関連する研究：アクセント記号復元

濁点付与とよく似たタスクに**アクセント記号復元**がある．これはアクセント記号を省略して書かれたフランス語やイタリア語のテキストに対して，書き手が元々意図していたアクセント記号付き表記を推定し，欠落したアクセント記号を補完するというタスクである．

2.1 単語ベースの手法

Yarowsky は文献 [8] で，アクセント記号復元をクラス分類問題として定式化した．この手法では，テキスト中の単語ごとにアクセントの推定が実施される．たとえば，フランス語の単語 *cote* が与えられたときに，それが *côte*（英訳：coast）であるのか，それとも *côté*（英訳：side）であるのかを分類器を用いて判定する．分類の素性には，アクセントを推定したい当該単語（target word）の情報と，その周辺文脈の情報が用いられる．周辺文脈の情報には target word とその左右に存在する単語とのコロケーションや，target word の周囲に存在する単語の情報などを使用する．

Yarowsky の手法では，テキスト内の各単語に対してそれぞれ独立にアクセントの推定が実施される．そのため，target word の周辺にある単語へのアクセント推定結果も target word 分類時にはいっさい参照しない．これに対して Simard は文献 [6] で，アクセント記号復元を系列ラベリングとして解く方法を提案している．

Yarowsky，Simard 両名の手法はいずれもアクセントの推定を単語単位で行う．そのため，上述の手法を日本語の濁点付与で使うには，まず未校訂資料を単語分割する必要がある．現在，歴史的資料の単語分割を高精度で行えるのは，形態素解析辞書**近代文語 UniDic** [9] や**中古和文 UniDic** [10] を用いた手法 [2] だけであるが，これらの辞書は校訂済みの資料を解析するために整備されており，未校訂の歴史的資料に対する解析精度は保証されない．また中世や近世の資料を単語分割するための手法はいまだなく，単語情報のアノテーションされたコーパスすら存在しない．

2.2 文字ベースの手法

使用可能なリソースが少ない言語でアクセント記号復元を行うために，Mihalcea は文献 [4] で，文字単位にアクセントを推定する手法を提案した．Mihalcea の手法は Yarowsky の手法と同様，アクセント記号復元をクラス分類問題として扱う．しかし，Yarowsky の手法とは異なり，分類はアクセントを推定したい当該文字（target char）ごとに実施される．また，指定した窓内の情報のみを周辺文脈情報として使う Windowing アプローチが採用されている．具体的には，target char から左右 N 文字の範囲（幅 $2N+1$ の窓（**図 2** 参照））内に存在する文字 1-gram のみ

[*4] 1874 年（明治 7 年）～1875 年（明治 8 年）の間に明六社から発行された啓蒙雑誌（全 43 号）

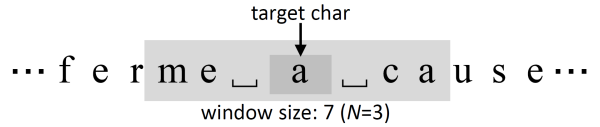


図 2 窓の例．ここでは target char “a” から左右 3 文字の範囲で窓を作成している（$N = 3$）

Fig. 2 Example of a window. Here, the window contains three characters to the left and right of the target char “a” ($N = 3$).

を分類の素性として使用する．Mihalcea はこの手法でルーマニア語のテキストに 99%以上の正解率でアクセント記号を付与することに成功している．

Mihalcea の手法では，Yarowsky と同じく分類を各文字についてそれぞれ独立に行い，周囲の文字へのアクセントの推定結果はいっさい参照しない．これに対し Wagacha らは文献 [7] で，テキストを左から右へ順番に走査し，貪欲法的にアクセントを推定していく手法を提案している．さらに，Wagacha らは文字 1-gram の代わりに文字 3-gram を素性に使うことも検討している．バントゥー語のテキストに対して，分類の素性に文字 1-gram を使用した場合の正解率は 77.5%であったが，文字 3-gram に変更することで 91.4%の正解率を達成している．

本論文でも Yarowsky や Mihalcea，Wagacha らと同じく，濁点付与をクラス分類問題として扱う．また提案手法では，Mihalcea，Wagacha らと同様，単語の情報をいっさい用いない文字ベースの処理を採用し，単語分割を必要としない．分類には Mihalcea と同じく，各文字ついてそれぞれ独立に分類を行う手法を採用した．こういった分類手法には，部分的にアノテーションされたテキストからでも学習が行えるという利点がある [5]．そのため提案手法では，部分的に濁点の付いた未校訂資料も学習に利用するが，こうした試みは Yarowsky や Mihalcea の手法にはなかったものである．

3. 提案手法

本論文では，濁点の自動付与のタスクを文字単位のクラス分類問題として定式化する．具体的には，校訂前のテキスト中に存在する濁点の付く可能性のある文字（**清濁曖昧文字**）を濁点文字と濁点の付かない文字のいずれかに分類する問題を扱う．本論文では，清濁曖昧文字として平仮名と踊字である**くの字点**（／＼，／″＼）だけを扱い，片仮名は扱わないこととした[*5]．そのため，濁点付与の対象となる文字は**図 3** にあげた全 22 種類である．

[*5] 漢字片仮名交じり文を除いて，基本的に片仮名は外来語や固有名詞等の限られた語の表記にしか用いられていない．実際，明六雑誌コーパス [13] 内の延べ語数は約 18 万であるが，その中で片仮名語の数はわずか 567 であり，さらにその中で人手で濁点付与が行われているものはわずか 6 例であった（2012 年 10 月段階のデータ）．また，片仮名で書かれた外来語や固有名詞に関しては濁点を付けるか否かの判定が人間でも困難な場合が多い．

か，き，く，け，こ，	さ，し，す，せ，そ，
た，ち，つ，て，と，	は，ひ，ふ，へ，ほ，
ゝ，／＼	

図 3 濁点付与の対象となる文字（清濁曖昧文字）

Fig. 3 Target characters for labeling voiced consonant mark (mark-less character).

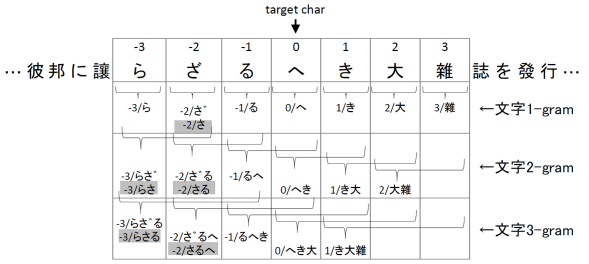


図 4 文字 n-gram（$N = 3$）

Fig. 4 Character n-gram ($N = 3$).

提案手法では，Windowing アプローチによる文字ベースの処理を採用しており，target char の周辺文字列の表層的な情報だけを分類の素性として使用する．

また提案手法では，清濁曖昧文字の分類を各文字についてそれぞれ独立に実施する方法を採用している．

3.1 分類に使用する素性

3.1.1 文字 *n*-gram

提案手法では，target char とその左右の N 文字（幅 $2N+1$ の窓）を 1 個の事例と見なし，1 個の素性ベクトルを作成する．各素性には**図 4** にあげたような文字 n-gram を使用する．また，Wagacha らの研究では，分類の素性に文字 3-gram を利用することの有効性が確認されており，本論文でも $n = 2$ 以上の文字 n-gram を素性に使うことを検討する．ただし，アルファベットが全 26 文字であるのに対し，日本語において使用される文字の総数はその 100 倍以上もある．そのため，高次の文字 n-gram を素性に使う場合には，何らかのスムージングが必要となる．そこで文字 n-gram だけでなく，文字 $(n-1)$-gram〜文字 1-gram も素性に加えることにした．つまり，$n = 3$ の場合，窓内に存在する文字 3-gram だけでなく，文字 2-gram と文字 1-gram も分類に使用する．

素性関数は以下のとおり．

$$\phi_j(window) = \begin{cases} 1: window \text{ 内の位置 } pos \text{ に} \\ \quad \text{文字列 } c_{pos}c_{pos+1}\cdots \text{ が出現} \\ 0: \text{それ以外} \end{cases} \quad (1)$$

ここで，pos は target char からの相対位置，c は文字を表している．また，$window$ は文字列 $c_{-N}, c_{-(N-1)}, \cdots, c_0, \cdots, c_{N-1}, c_N$ であり，c_0 が target char である．

3.1.2 疑似濁点無表記文字 *n*-gram

歴史的資料は表 1 で示しているように，完全に無濁点に

なっているとは限らない．ところどころには濁点が付いた状態の資料もある．そして，どの文字に濁点を付け，どの文字に付けないかは書き手によって様々である．そのためスパース性を避け，濁点付与の再現率を上げるためには，分類器の学習時にも，推定時にも，資料中の濁点をすべて外し，無濁点にしておくべきである．

しかしながら，あらかじめ施されていた濁点は分類の際の証拠として有効な場合もあると考えられる．そのため，提案手法では事前に資料を無濁点にはしない．そのかわり，素性ベクトルを作成する際は文字 n-gram 中の濁点を一部～すべて外した文字 n-gram の素性も同時に発火させることにした．たとえば，位置 *pos* における文字 3-gram が「がぎぐ」であるとするなら，“*pos*/がぎぐ” だけでなく，*pos*/がぎく，*pos*/がきぐ，*pos*/かぎぐ，*pos*/がきく，*pos*/かぎく，*pos*/かきぐ, *pos*/かきくの素性も同時に値 1 とする．これらの素性は図 4 ではハイライトで示してある．

3.2 訓練用事例の作成手順

3.2.1 校訂済み資料からの事例抽出

提案手法において，訓練用事例は基本的に，濁点無表記文字を含まない校訂済みコーパスから作成する．訓練用コーパス中の濁点文字から**濁点文字の事例**，濁点の付いていない文字から**濁点の付かない文字の事例**を作成する．これらの事例は清濁曖昧文字の分類に利用するため，事例作成時，濁点文字の事例では target char の濁点を外しておく．校訂済みコーパスからの訓練用事例の作成手順は以下のとおり．

Step 1：訓練用コーパスから訓練用文字（**図 5** 参照）を 1 つ取り出す．

Step 2：Step 1 で取り出した文字とその左右 N 文字を合わせて 1 つの事例と見なし，素性ベクトルを作成する．この際，Step 1 で取り出した訓練用文字が濁点文字であれば，その濁点を外した後，素性ベクトルを作成する．

Step 3：Step 1 で取り出した訓練用文字を事例のクラスラベルとする．

3.2.2 未校訂資料からの事例抽出

提案手法では，資料中の清濁曖昧文字をそれぞれ独立に分類する手法を採用した．これは未校訂資料からでも濁点の付いた文字から訓練用事例を抽出し，学習に使うことができるからである [5]．

提案手法では，窓内に target char 以外に清濁曖昧文字があっても，それらの文字に対して行われた分類の結果は当該分類からはいっさい参照しない．たとえば「能はさる（アタワザル）」に濁点を付与する場合，「は（ワ）」に対して行った分類の結果は「さ（ザ）」を分類する際の素性ベクトルにいっさい反映されない．そのため target char のラベルさえ明確ならば，窓内で他の文字のラベルが曖昧であったとしても訓練用事例が抽出でき，学習に使用可能である．つまり，「及はざる（オヨバザル）」のように濁点無表記を含んだ箇所からも「ざ」のように曖昧性のない “濁点文字” ならば，3.2.1 項で述べた手順で濁点の付く文字の事例を抽出し，学習に使うことが可能である．反対に，未校訂資料中の濁点の付いていない文字は濁点無表記の可能性があり，学習には使うことができない．

か，き，く，け，こ，	さ，し，す，せ，そ，
た，ち，つ，て，と，	は，ひ，ふ，へ，ほ，
ヽ，／＼（くの字点）	
が，ぎ，ぐ，げ，ご，	ざ，じ，ず，ぜ，ぞ，
だ，ぢ，づ，で，ど，	ば，び，ぶ，べ，ぼ，
ヾ，／″＼（くの字点）	

図 5 訓練用文字一覧
Fig. 5 List of characters during training.

Simard のような系列ラベリング手法では，すべての濁点無表記文字に濁点が付与された校訂済み資料しか学習に利用できない．しかし提案手法ならば，上記の方法を使って未校訂資料からでも学習が行えるため，学習に使える校訂済み資料が少ない中世や近世の資料の濁点付与にも応用可能である．

3.3 句読点の扱い

句読点の使い方の標準を定めた句読法案が出されたのは 1906 年（明治 39 年）であり，それまでは句点と読点の使い方が正式に定まっていなかった．そのため図 1 のように句読点がいっさい含まれていない資料もあれば，読点のみで句点がいっさいない場合もある．また現代の書き言葉では，句点で文の区切りとするが，歴史的資料の中では読点を使って文境界や，段落の切れ目を表している場合すらある．

このように，句読点の使用が確立する以前に書かれた資料の中では，句読点の使い方が一貫していない．そこで提案手法では，資料中に句読点「、。」が含まれる場合には，事前にそれらを特殊記号 ⟨PUNC⟩ で置き換え，区別しないことにした．

4. 濁点付与の性能評価実験

提案手法の有効性を検証するために濁点付与の性能評価実験を行った．本実験では未校訂の**近代文語論説文**（**図 6** のような文体）を対象とし，濁点付与の適合率と再現率を調べた．

4.1 実験に使用したコーパス

本実験では，以下の近代語コーパスを利用する．

UniDicMLJ-TRAIN： 近代文語 UniDic v1.1 のコスト算出に用いられたコーパス．形態素解析済みコーパ

夫維新以來賢材も輩出し百度も更張し官省寮司より六十餘縣に至るまて既に昔日の日本に非す其善政美擧も屈指に暇あらさるなり然るに退て熟々之を考ふれは百端未た脱垢の地に至らさる事のみにして善政あれとも民其澤を蒙らす美擧あれとも得失相償はさる等の事多し

図 6 近代文語論説文の例．この文章は，明六雑誌 1 号からの抜粋である．漢字片仮名交じり文で書かれた文章を漢字平仮名交じり文に直している．また，清濁曖昧文字を太字にして示している

Fig. 6 Examples of sentences in near-modern literary style. This text is an excerpt from Meiroku Zasshi, Vol.1. We replaced all Katakana characters with Hiragana characters. Bold characters are mark-less characters.

スであり，校訂も実施済みである．ただし量が少なく，濁点無表記のような原文の情報もコーパス中には保持されていない．

SUN-TRAIN, SUN-TEST：近代語の大規模コーパスである太陽コーパス（2011 年 1 月段階のデータ）から 9 割を訓練用コーパスとして利用する．実際には，太陽コーパスの 1895 年 5 号，1901 年 5 号，1909 年 5 号，1917 年 5 号，1925 年 5 号を評価用コーパスとして別に分け（SUN-TEST），残りを学習に利用した（SUN-TRAIN）．太陽コーパスは校訂済みコーパスであるが，タグを使って原文の情報も保持されている．ただし，単語情報のアノテーションまでは行われていない．訓練用コーパスには校訂済みの本文を使い，評価用コーパスにはタグから再現した原文を使用する．

NF-TRAIN, NF-TEST：提案手法の利点として，3.2.2 項で述べたように，目的の時代や文体の校訂済み資料が少量しか用意できない場合でも，未校訂資料から濁点文字の事例を取り出し，当初の訓練用の事例に追加可能なことがあげられる．そこで，現在コーパス化の作業が進められている明治期の雑誌**国民之友**から，2011 年 3 月の段階で濁点付与が実施されていた 1887 年 10 号，1888 年 20 号，1888 年 30 号，1888 年 36 号を評価用コーパスとし（NF-TEST），残りの未校訂分を学習に利用する（NF-TRAIN）．国民之友も太陽コーパスと同様，タグを使って原文情報が保持されており，単語情報のアノテーションまでは行われていない．評価用コーパスにはタグから復元した原文を使用する．

M6-TEST：明六雑誌コーパス [13]（2011 年 12 月段階のデータ，濁点付与実施済み）も評価用コーパスとして利用する．評価には SUN-TEST や NF-TEST と同じくタグから復元した原文を使用する．ただし，明六雑誌の原文はほぼすべての記事が漢字片仮名交じり文で記述されている．そのため，ここではすべての片仮名文字を平仮名文字に置き換えて使用する．

明治期においてはいまだ句読点の使い方が定まっておらず，太陽コーパスや国民之友でも明確な文境界を定めることは難しい．そこで今回，この 2 つのコーパスは学習時において，段落単位で使用することにした．ただし，UniDicMLJ-TRAIN は近代文語 UniDic のコスト計算に使用されたコーパスであり，近代文語 UniDic を用いた形態素解析は文単位で行われるため，UniDicMLJ-TRAIN では文境界が明確にされている．そこで，UniDicMLJ-TRAIN は文単位で使用することにした．また，評価用のコーパスも実験設定を実際の濁点付与の状況に近づけるため，段落単位で使用する．訓練・評価用コーパスの各文・各段落の先頭と末尾には，それぞれ文頭，文末を表す特殊文字 ⟨BOS⟩ と ⟨EOS⟩ を追加した．

この実験では，文語を扱うが，UniDicMLJ-TRAIN 以外の雑誌コーパスには，口語で書かれた記事も含まれている．そのため，タグの情報を基に，口語の記事や引用は訓練用コーパス・評価用コーパスのいずれからもすべて除外した[*6]．口語文を除外した各コーパスの文数・段落数，総文字数と未校訂状態での濁点文字数の内訳を**表 2** に示す．また，訓練用事例と評価用事例の内訳を**表 3** と**表 4** に示す．NF-TRAIN において濁点の付かない文字の事例数が 0 になっているが，これは 3.2.2 項でも述べたとおり，未校訂資料からは濁点文字の事例は抽出可能であるが，濁点を付けない文字の事例を取り出すことができないからである．

4.2 使用した分類器と学習アルゴリズム

今回は，太陽コーパスのような大規模なコーパスからでも高速かつ高精度に分類器の学習を行うため，分類器として線形識別器を使用する．また，学習には Passive-Aggressive（PA）[1] を採用した．PA アルゴリズムは，各訓練用事例に対し，素性の重みをそのつど更新するオンライン学習アルゴリズムの 1 つである．ここでは，PA に重みの更新速度を制御する変数 C を導入した PA-I を使用する．

分類は以下の式により行われる．

$$\hat{y} = \underset{y \in \{\, c_0, c_0+^{\circ} \,\}}{\arg\max} \mathbf{w} \cdot \Phi'(window, y) \tag{2}$$

$$\phi'_k(window, y) = \begin{cases} 1 : \text{“}\phi_j(window) = 1\text{”} \cap \text{“ラベル} = y\text{”} \\ 0 : \text{それ以外} \end{cases}$$

ここで，$\hat{y}$ は分類結果，$\mathbf{w}$ は重みベクトル，Φ' は素性ベクトル $[\phi'_1, \phi'_2, \cdots, \phi'_k, \cdots]$ をそれぞれ表している．y はラベ

[*6] 文献 [14] によると，太陽コーパス内の文語体で書かれた記事数と口語で書かれた記事数の割合はほぼ 1 : 1 であるが，口語体で書かれた記事においては濁点無表記の文字が非常に少ない．文献 [12] によれば，太陽コーパスの口語記事内では濁点文字使用率が 99.7%となっている．これは言文一致によって口語体が普及するとともに，濁点を使った濁音の明示化が普及したことによる．そのため，近代の口語体に関しては濁点付与を行う必要がほとんどないと考えられる．

表 2 コーパス内の文数・段落数，総文字数と未校訂状態での濁点文字数の内訳
Table 2 Number of sentences, paragraphs, characters and marked characters before textual criticism in each corpus.

	文数	段落数	総文字数	未校訂状態での濁点文字数
UniDicMLJ-TRAIN	20,330	-	604,966	-
SUN-TRAIN	-	70,084	6,380,398	-
NF-TRAIN	-	6,510	1,437,845	-
SUN-TEST	-	6,316	619,357	20,503
NF-TEST	-	868	172,780	3,127
M6-TEST	-	1,450	252,232	4,596

表 3 訓練用事例の内訳
Table 3 Number of instances in each training corpus.

訓練用コーパス	濁点文字	濁点の付かない文字	計
UniDicMLJ-TRAIN	26,123	110,974	137,097
SUN-TRAIN	208,099	962,580	1,170,679
NF-TRAIN	24,195	0	24,195

表 4 評価用事例の内訳
Table 4 Number of instances in each test corpus.

評価用コーパス	濁点文字（濁点無表記）	濁点の付かない文字	計
SUN-TEST	899	92,803	93,702
NF-TEST	3,842	25,418	29,260
M6-TEST	6,219	39,314	45,533

ルであり，ラベルの集合には図 5 を使用する．また，c_0 と $c_0 + ^{\circ}$ はそれぞれ target char と，target char に濁点を付けた文字を表している．ϕ_j は 3.1.1 項で定義した素性関数である．

モデルの学習は**図 7** のようにして行う．ここで，y'_t は訓練事例 $(window_t, y_t)$ のラベル y_t が濁点文字の場合，y_t から濁点を外した文字，清音文字の場合，y_t に濁点を付けた文字をそれぞれ表す．

本実験ではイテレーション回数をすべて 20 回に設定し，パラメータ C は訓練用コーパスを用いた 5-fold のクロスバリデーションで平均 F 値が最大になる値に定めた．

4.3 比較手法

4.3.1 ベースライン

ベースラインとして，文字 3-gram 言語モデルを作成し，尤度が最大になるように濁点を付与する手法を設定した．言語モデルの作成には Palmkit v1.0.32 [*7] を使用し，スムージングには Witten-Bell を用いた．また尤度最大の文字列は Viterbi アルゴリズムによって求める．

4.3.2 辞書ベースの手法

文字ベースの提案手法に対して，単語単位で濁点付与を行う手法を比較に用意した．この手法では以下の手順で近代文語 UniDic v1.1 を拡張し，拡張した辞書による形態素解析結果から濁点付与を行う．形態素解析器には MeCab v0.98 を使用する．

Step 1：近代文語 UniDic 中のすべての語から濁点を取り除く．ただし，各語のフィールド中には濁点を外す前の表記を保存しておく．

Step 2：無濁点にした UniDicMLJ-TRAIN を用いて Step 1 で作成した辞書のコストを計算する [*8]．

Step 3：各語のフィールド中に残しておいた濁点付きの表記から，濁点の一部～すべてが補われた書字形を復元し，辞書に追加する．ただし，このとき追加する語のコストは，無濁点の場合のコストと同一とする．

この辞書を用いることで，濁点無表記を含んだ文でも形態素解析が行える．また，各語のフィールド中には濁点付きの表記が保存されているため，形態素解析の結果から濁点付きの表記を復元できる．

辞書ベースの手法では，形態素解析済みのコーパスから

Passive-Aggressive-I アルゴリズム

INPUT: Training set $T = \{(window_t,\ y_t)\}_{t=1}^{|T|}$,
Number of iterations I and Parameter C
OUTPUT: $\mathbf{w}$

Initialize: $\mathbf{w} = \mathbf{0}$
for $i \leftarrow 1$ to I do
  foreach $(window_t, y_t) \in T$ do
    $\gamma = \mathbf{w} \cdot \Phi'(window_t,\ y_t) - \mathbf{w} \cdot \Phi'(window_t,\ y'_t)$
      $(y'_t \neq y_t,\ y'_t \in \{c_0,\ c_0 + ^{\circ}\})$
    if $\gamma < 1$ do
      $\tau_t = min\left(C,\ \frac{1-\gamma}{||\Phi'(window_t,\ y_t) - \Phi'(window_t,\ y'_t)||^2}\right)$
      $\mathbf{w} \leftarrow \mathbf{w} + \tau_t(\Phi'(window_t,\ y_t) - \Phi'(window_t,\ y'_t))$
    end if
  end foreach
end for
return $\mathbf{w}$

図 7 Passive-Aggressive-I
Fig. 7 Passive-Aggressive-I.

[*7] http://palmkit.sourceforge.net/

[*8] どの文字から濁点を脱落させるかは書き手によって異なるため，できるだけ多くの濁点無表記のパターンがコーパス中に備わっていることが望ましい．そのため，あえてコーパスを無濁点にしてコストの再計算に利用した．

表 5 性能の内生的な評価：濁点付与性能
Table 5 Performance of intrinsic evaluation: performance of labeling voiced consonant mark.

評価用コーパス	訓練用コーパス	手法	Prec. [%]	Rec. [%]	F
SUN-TEST	UniDicMLJ-TRAIN	ベースライン	33.7	86.5	48.5
		辞書ベース	50.9	91.8	65.5
		提案手法	54.7	85.2	66.6
		提案手法（Gold）	54.4	85.0	66.3
	UniDicMLJ-TRAIN + SUN-TRAIN	ベースライン	54.0	95.2	68.9
		提案手法	**71.2**	**97.0**	**82.1**
		提案手法（Gold）	**71.2**	**97.0**	**82.1**
NF-TEST	UniDicMLJ-TRAIN	ベースライン	90.6	95.7	93.1
		辞書ベース	93.3	96.5	94.9
		提案手法	95.1	94.5	94.8
		提案手法（Gold）	95.2	94.8	95.0
	UniDicMLJ-TRAIN + SUN-TRAIN	ベースライン	93.8	98.1	95.9
		提案手法	96.0	98.3	97.1
		提案手法（Gold）	**96.2**	98.3	**97.2**
	UniDicMLJ-TRAIN + NF-TRAIN	提案手法	89.3	97.6	93.2
	UniDicMLJ-TRAIN + NF-TRAIN + SUN-TRAIN	提案手法	95.7	**98.6**	97.1
M6-TEST	UniDicMLJ-TRAIN	ベースライン	87.1	94.0	90.4
		辞書ベース	90.1	95.9	92.9
		提案手法	93.4	92.4	92.9
		提案手法（Gold）	93.4	92.2	92.8
	UniDicMLJ-TRAIN + SUN-TRAIN	ベースライン	90.5	98.1	94.2
		提案手法	94.7	98.1	96.4
		提案手法（Gold）	**94.9**	**98.2**	**96.5**

でしか学習が行えないという欠点がある．現時点で，近代文語論説文の形態素解析済みコーパスはUniDicMLJ-TRAINと明六雑誌コーパス（2012年10月段階のデータ）以外に存在しない．また，濁点付与の性能が形態素解析の精度に依存する．そのため，濁点付与の性能を上げるためには，学習用の形態素解析済みコーパスを新しく整備するか，辞書のエントリを増やす必要がある．しかし，これらの作業は一般に濁点付与よりもコストが高い．

4.4 提案手法の Gold Standard

提案手法では，各分類で周辺の分類結果はいっさい参照しない．そのためたとえば，「つゝましき（正解：つゝましき）」「つゝくる（正解：つゞくる）」「一人つゝ（正解：一人づゝ）」のように，「ゝ」が「つ」の分類結果に依存するような場合，両方に濁点を付けるなどして濁点付与に失敗する恐れがある．そして，このような場合には，Simardのような系列ラベリングが有効になる．

しかし提案手法には，未校訂資料も学習に使用できるという利点がある．反対に，系列ラベリングではこれが行えない．

そこで今回，訓練用事例・評価用事例を作る際にあらかじめ窓内にあるtarget char以外の文字の濁点を正しく付けておく手法を提案手法のGold Standardとして設けた[*9]．これにより，系列ラベリングの上限を知ることができ，提案手法と比較することができる．

4.5 濁点付与性能評価実験

各手法を用いてそれぞれの評価用コーパスに濁点付与を行い，評価を行った．ここでは濁点付与の適合率，再現率，F値で評価した．提案手法のパラメータは，窓幅：7（$N = 3$），素性に用いるn-gramの最大長を3（$n = 3$）に設定した[*10]．結果を**表 5** に示す．

[*9] この手法では，target charの周辺文脈に濁点無表記がいっさい含まれない設定で分類を行うため，疑似濁点無表記の素性は使用しない．また，疑似濁点無表記の素性を追加した場合，わずかに性能が低下することを確認している．

[*10] 近代文語論説文の場合，nの値を1から大きくしていくと，性能は$n = 3$で大体頭打ちとなる．また，nが3よりも大きくなると再現率が低下し始め，その分，適合率が若干高くなる．Nに関しても同様で，Nを3よりも大きくした場合，適合率がわずかに上昇し，再現率がわずかに低下する．ただし，F値で比較すると性能に変化は見られなかった．Nやnの値を大きくすると，素性数が増え，モデルが肥大化し，学習にも時間がかかる．そのため，$N = 3$，$n = 3$を用いるのが大体妥当であると考えられる．

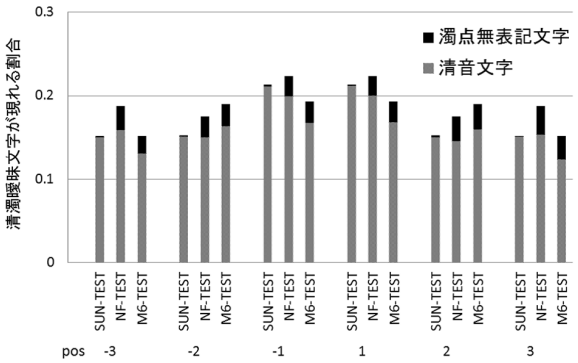


図 8 評価用事例の window 内の各位置に清濁曖昧文字が現れる割合
Fig. 8 The occurrence ratio of mark-less characters at each position of window in the test instances.

$$Precision = \frac{\text{正しく濁点を付けた文字数}}{\text{濁点を自動付与した文字数}} \times 100\,[\%]$$

$$Recall = \frac{\text{正しく濁点を付けた文字数}}{\text{評価用コーパス中の濁点無表記文字数}} \times 100\,[\%]$$

$$F = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

訓練用コーパスを UniDicMLJ-TRAIN でそろえた場合，再現率においては，辞書ベースの手法の性能が最も高かった．次点はベースライン手法であり，提案手法は再現率が最も低かった．しかし適合率を比べると提案手法の性能が最も高い．そのため F 値で比較すると，提案手法は辞書ベースと同等の性能であった．またベースライン手法は再現率は高いものの，適合率が他と比較して低い．ゆえに F 値の比較においては総じて最下位の性能であった．

提案手法は文字ベースの処理を採用している．そのため辞書ベースの手法に比べて，低コストで訓練用コーパスを追加できるという利点がある．訓練用コーパスとして SUN-TRAIN を追加したとき，提案手法は，適合率，再現率，F 値のすべてで辞書ベースの手法よりも高い性能を示した．このように比較的簡単に性能を上げられるという点において，辞書ベースの手法に対する提案手法の優位性が確認できた．

提案手法を Gold の結果と比較すると，提案手法でもほぼ上限の性能が得られていることが分かる．また実際，提案手法と Gold の濁点付与の結果の母比率の差を McNemar 検定で比較しても，有意水準 5%では両手法に有意な差はみられなかった．これは提案手法の window と Gold の window の間に違いがほとんどないためだと考えられる．**図 8** に評価用事例の window 内にどれだけ清濁曖昧文字が現れるのか，コーパスごとに割合を調査した結果を示す．これを見ると，どの位置にも大体 15〜20%の割合で清濁曖昧文字が現れていることが分かる．しかし，その大部分が濁点無表記文字ではない．つまり，提案手法の window は Gold の window とほぼ同じであり，そのため性能もほとんど同じになっているのだと考えられる．また，Gold は系列ラベリングの上限であるから，少なくとも近代文語論説文においては提案手法でも系列ラベリングとほぼ同等の性能が出せることが示せた．

表 6 NF-TEST におけるエラーの分布
Table 6 Distribution of error in NF-TEST.

	エラーの数
E1: サ変動詞と，ザ変動詞	23
E2: 当時，語形上揺れがあった語（e.g.,「願わくば」と，「願わくは」）もしくは濁点を付けても付けなくてもどちらでもよさそうな語（e.g.,「結び」と，「結ひ」,「出て」と，「出で」）	23
E3: 格助詞「が」・接続助詞「が」と，終助詞「か」・並列助詞「か」	20
E4: 打消しの助動詞「ず」と，サ変動詞「す」	3
E5: 接続助詞「ば」と，係助詞「は」	2
アノテーションミス（濁点無表記文字の見落とし）	13
その他	16

提案手法のエラー分析を行ったところ，訓練用コーパスにおいて未知の事例や，「いなどいひて（正解：いな（否）といひて）」のように部分文字列にマッチする単語に引かれて濁点付与に失敗するほか，以下にあげた語と語の間で失敗する傾向がみられた．

E1：サ変動詞と，ザ変動詞
E2：当時，語形上揺れがあった語（e.g.,「願わくば」と，「願わくは」）もしくは濁点を付けても付けなくてもどちらでもよさそうな語（e.g.,「結び」と，「結ひ」,「出て」と，「出で」）
E3：格助詞「が」・接続助詞「が」と，終助詞「か」・並列助詞「か」
E4：打消しの助動詞「ず」と，サ変動詞「す」
E5：接続助詞「ば」と，係助詞「は」

NF-TEST で生じたエラー[*11]からランダムに 100 件取り出し，エラーの種類ごとに集計した結果を**表 6** に示す．表 6 を見ると，エラー項目の E1〜E3 がそれぞれエラーの大体 1/5 ずつを占めていることが分かる．また，取り出した 100 件のエラーの中から，アノテータがとりこぼしていた濁点無表記文字が 13 件見つかった[*12]．このことから，人手でとりこぼしがちな濁点無表記文字も，提案手法を使うことでカバーできることが確認できた．

エラー項目の E3〜E5 は文の意味にかかわる項目である．そのため，文字単位で処理を行う提案手法や，語の連接コストに基づく辞書ベースの手法では，濁点を付けるべきか否かが曖昧で難しくなっていると考えられる．実際，提案手法と辞書ベースの手法のエラーを比較しても，上記の濁点

[*11] 訓練用コーパスは UniDicMLJ-TRAIN+SUNTRAIN を使用．
[*12] 4.1 節でも述べたとおり，国民之友（NF-TEST）は整備途中のコーパスであり，整備が完了している太陽コーパス（SUN-TRAIN, SUN-TEST）や明六雑誌コーパス（M6-TEST）に比べると，いまだ濁点を付けるべき文字を見落としている場合がある．

付与誤りの傾向に大きな差はみられなかった．SUN-TEST における適合率が総じて低いのは，上のようなエラーがつねに生じてしまうが，そのエラーの数に対して SUN-TEST 中の濁点無表記文字の数（濁点を付けて正解の箇所）が極端に少ないためである．逆にいうと，表 4 からも分かるように，NF-TEST や M6-TEST は SUN-TEST に比べて濁点無表記文字の割合が高く，濁点付与が比較的容易な事例が多く含まれている．そのため，（NF-TEST が整備途中であることを差し引いても）SUN-TEST より高い性能が得られていると考えられる．

辞書ベースの手法では，以下の例のように形態素解析に失敗し，その結果，濁点付与にも失敗することがあった．

人 | の | 氣附:名詞（キヅケ）| が:助詞 | さる:連体詞 | 所ろ |、

（**正解：**人 | の | 氣附か：動詞 | ざる：助動詞 | 所ろ |、）

反対に，辞書ベースの手法が有利に働いた例として，「ゆへ（故）」がある．「ゆへ」は歴史的仮名遣に照らし合わせると「ゆゑ」と表記されるべきである．そのため，校訂済みコーパスでは，「ゆゑ」に表記が直されている．つまり，校訂済みのコーパスで学習を行った提案手法では，「『ゆへ』の『へ』には濁点を付けない」ということを学習できていない．これに対し，近代文語 UniDic では仮名遣のバリエーションもある程度考慮されている．そのため，この事例に関しては，濁点付与に失敗することがなかった．

NF-TRAIN を UniDicMLJ-TRAIN に追加することで，他の手法に比べて低かった提案手法の再現率が大いに向上した．その結果，ベースラインや辞書ベースの手法よりも再現率が高くなった．またその反面，適合率は約 6%低下した．これは，以下の 2 つのことが原因だと考えられる．

(1) UniDicMLJ-TRAIN と NF-TEST の間には，表記状態およびドメインの違いがある．

(2) 提案手法には，濁点を付けるべきか否かが曖昧で，判断に揺れる事例がある．

NF-TEST の文体は UniDicMLJ-TRAIN と同じ近代文語論説文ではあるが，表記の状態が異なっている．NF-TEST は未校訂の状態だが，UniDicMLJ-TRAIN の表記は校訂済みの比較的きれいな形である．また，国民之友は総合雑誌であるため，様々なドメインの記事を含んでいる．そのため，UniDicMLJ-TRAIN だけで学習したモデルでは，NF-TEST 中に未知の事例が出現することが起こりうる．その場合，UniDicMLJ-TRAIN だけで学習したモデルでは，基本的に濁点は付けない．これは表 3 で示したとおり，濁点を付けない文字の事例の方が濁点を付ける文字の事例よりも多いため，分類の曖昧な事例が濁点を付けない方へと振れるためである．しかし，NF-TEST と同じ国民之友からとってきた NF-TRAIN を追加することで，UniDicMLJ-TRAIN で未知であった事例にも濁点を付けることが可能になる．ただし，NF-TEST からは負例を抽出しないため，先に述べた終助詞「か」と格助詞「が」のように提案手法で分類が難しい個所は，UniDicMLJ-TRAIN で未知で NF-TRAIN で既知のものがすべて濁点を付けるようになってしまう．これが適合率低下の原因であると考えられる．また，UniDicMLJ-TRAIN+NF-TRAIN へさらに，国民之友と同じ総合雑誌である SUN-TRAIN を追加すれば，適合率の低下を抑えつつ Recall を上げることできる．

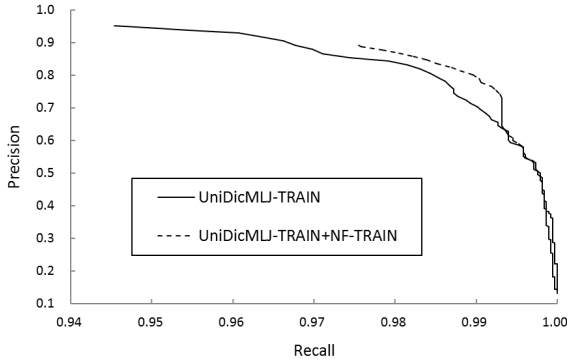


図 9 提案手法にしきい値を導入した際の UniDicMLJ-TRAIN と UniDicMLJ-TRAIN+NF-TRAIN の比較

Fig. 9 Comparing of UniDicMLJ-TRAIN and UniDicMLJ-TRAIN+NF-TRAIN when introducing threshold value for our method.

また，提案手法にしきい値を導入し，UniDicMLJ-TRAIN のみで学習したモデルと NF-TRAIN を追加して学習したモデルを比較した．具体的には，濁点付与の際，濁点を付ける場合のスコアと付けない場合のスコアの差の絶対値がしきい値未満の場合，分類結果を信用せず必ず濁点を付ける．この設定でしきい値を 0 から順番に上げてプロットした結果の PR 曲線を**図 9** に示す[*13]．この PR 曲線から，UniDicMLJ-TRAIN のみの場合で NF-TRAIN を追加した場合と同等の再現率 97.6%を実現しようとすると，NF-TRAIN を追加した場合よりも適合率が約 4.3%低くなることが分かった．また曲線全体を比較しても，NF-TRAIN を追加した方が適合率を高く維持したまま再現率を向上させることができている．

表記整理作業において，濁点の自動付与には，人手での濁点無表記文字の見落としをカバーできることが一番期待されている．そのため，適合率よりも再現率が重視される．したがって，再現率の向上がみられた点において，未校訂資料を学習に利用することの有効性が確認できた．

5. 形態素解析性能の改善度の比較

近代文語 UniDic は本来，校訂済みの文を解析するために整備されている．そのため，未校訂の資料の形態素解析に利用しても，結果の精度は高くない．そこで提案手法を形態素解析の前処理に用いることで，形態素解析の性能がどれほど改善できるか調査した．

[*13] 各曲線の最左点がしきい値を使用しない場合（しきい値：0）を表している．

表 7 性能の外的な評価：形態素解析結果の一致率
Table 7 Performance of extrinsic evaluation: agreement rate of morphological analysis result.

	単語分割			品詞認定			語彙素認定		
手法	Prec. [%]	Rec. [%]	F	Prec. [%]	Rec. [%]	F	Prec. [%]	Rec. [%]	F
ベースライン	89.84	93.68	91.72	85.85	89.53	87.65	81.37	84.85	83.08
辞書ベース	**92.79**	95.65	**94.20**	**90.21**	**92.99**	**91.58**	**86.09**	88.75	**87.40**
提案手法	92.72	**95.67**	94.17	90.10	92.98	91.52	86.02	**88.77**	87.37

ただし，前章で評価に利用していた太陽コーパスや国民之友はいずれも濁点無表記のような原文情報は保持しているが，単語の情報まではアノテーションされていない．また，UniDicMLJ-TRAIN は近代文語 UniDic の学習に利用されたコーパスであるため，単語の情報のアノテーションは行われているが，原文情報までは保持していない．そこで今回，現在唯一，単語の情報がアノテーションされ，かつ濁点無表記や誤字・脱字といった原文情報も保持した明六雑誌コーパスの 2012 年 10 月段階のデータ（XML ファイル）[*14]を評価に利用する．評価は M6-TEST と同じく，文語記事のみを対象とし，タグから復元した原文を段落単位で使用する．また，実際の原文は 4.1 節で述べたように漢字片仮名交じり文であるが，2012 年 10 月段階のデータではすべて漢字平仮名交じり文に人手修正されている．そこで，M6-TEST で片仮名文字をすべて平仮名文字に統一していたのと同様，ここでも残っていた片仮名文字はすべて平仮名に統一した．

評価は単語分割，品詞認定，語彙素認定の 3 段階で行う．それぞれの段階における適合率・再現率・F 値を調査した．適合率・再現率・F 値の式は文献 [2] と同じものを使用する．

提案手法は UniDicMLJ-TRAIN + SUN-TRAIN で学習を行ったモデルを使用する．形態素解析には MeCab v0.98 + 近代文語 UniDic v1.1 を利用した．また，原文を通常の近代文語 UniDic で解析した場合（ベースライン）と，前処理に提案手法を使用せず，4.3.2 項で述べた辞書ベースの手法で濁点付与と形態素解析を同時に実施した場合とも比較を行った．ただし，近代文語 UniDic v.1.1 にはもともと少数であるが，濁点無表記の書字形が含まれている．比較を公平に行うため，提案手法およびベースラインで使用する近代文語 UniDic からは，濁点無表記の書字形をすべて取り除いた．

結果を**表 7** に示す．この結果を見ると，提案手法を前処理として用いることで，形態素解析性能が向上することが分かる．また適合率で若干劣るものの，提案手法は辞書ベースの手法とほぼ同等の性能が出せることも分かった．提案手法の適合率が低くなったのは，部分文字列にマッチする単語に引かれて濁点を過剰に付けてしまった箇所で，後続の形態素解析が失敗するためだと考えられる．

6. おわりに

本論文では，識別学習を用いて濁点の自動付与を行う手法を提案した．提案手法では文字ベースの処理を採用しており，学習に形態素解析済みコーパスを必要としない．そのため，低コストでの性能向上が可能である．また提案手法を前処理に用いることで，未校訂の資料の形態素解析性能が向上することが分かった．さらに，太陽コーパスのような大規模コーパスを使用できなくても，提案手法ならば，未校訂の資料からでも訓練用事例を作り，濁点付与の再現率を高めることが可能だと分かった．

本論文では，近代文語論説文で評価実験を行った．今回は近代文語論説文を一括りに議論したが，文献 [12] によると，濁点文字の使用率は年代によって異なっている．そのため，年代ごとの濁点の使用傾向を反映することが今後の課題としてあげられる．これには単純な方法として図 9 を描いた際のように，適合率と再現率を調整するしきい値を年代ごとに設定して使うことが考えられる．また今後は近代だけでなく，中古や近世の資料の濁点付与にも取り組むことを考えている．そして，濁点付与に限らず，提案手法の枠組みを使い，「用い（モチイ）」「用ゐ（モチイ）」「用ひ（モチイ）」のような仮名遣の混用を正規化する問題も将来的には扱っていきたいと考えている．

提案手法が濁点付与に失敗しやすい事例として終助詞「か」と格助詞「が」があった．これは未校訂資料中において文境界が明確でないことが原因で生じたエラーである．そのため，文境界判定（句点付与）も今後の課題である．

本論文で採用したオンライン学習には，モデルの再学習時にもとの学習データを必要とせず，またメモリを大量に消費しないという利点がある．そこで今後，提案手法に能動学習を取り入れ，提案手法の出力に対し，人間が修正を行い，その結果を使ってモデルを再学習するといったユーザインタラクティブな濁点付与支援ツールの開発に取り組んでいきたいと考えている．

謝辞 本研究は，国立国語研究所の共同研究プロジェクト「統計と機械学習による日本語史研究」による研究成果の一部である．

[*14] http://www.ninjal.ac.jp/corpus_center/cmj/meiroku/

参考文献

[1] Crammer, K., Dekel, O., Keshet, J., et al.: Online Passive-Aggressive Algorithms, *JMLR*, Vol.7, pp.551–585 (2006).
[2] Kudo, T., Yamamoto, K. and Matsumoto, Y.: Applying Conditional Random Fields to Japanese Morphological Analysis, *Proc. 2004 Conference on Empirical Methods in Natural Language Processing* (*EMNLP 2004*), pp.230–237 (2004).
[3] Maekawa, K.: Balanced Corpus of Contemporary Written Japanese, *Proc. 6th Workshop on Asian Language Resources* (*ALR 2008*), pp.101–102 (2008).
[4] Mihalcea, F.R.: Diacritics Restoration: Learning from Letters versus Learning from Words, *Proc. 3rd International Conference on Intelligent Text Processing and Computational Linguistics* (*CICLing 2002*), pp.339–348 (2002).
[5] Neubig, G., Nakata, Y. and Mori, S.: Pointwise Prediction for Robust, Adaptable Japanese Morphological Analysis, *Proc. 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* (*ACL HLT 2011*), pp.529–533 (2011).
[6] Simard, M.: Automatic Insertion of Accents in French Text, *Proc. 3rd Conference on Empirical Methods in Natural Language* (*EMNLP '98*), pp.27–35 (1998).
[7] Wagacha, W.P., Pauw, D.G. and Githinji, W.P.: A Grapheme-Based Approach for Accent Restoration in Gĩkũyũ, *Proc. 5th International Conference on Language Resources and Evaluation* (*LREC 2006*), pp.1937–1940 (2006).
[8] Yarowsky, D.: DECISION LISTS FOR LEXICAL AMBIGUITY RESOLUTION: Application to Accent Restoration in Spanish and French, *Proc. 32nd Annual Meeting of the Assoc. for Computational Linguistics* (*ACL 94*), pp.88–95 (2004).
[9] 小木曽智信，小椋秀樹，近藤明日子：近代文語文を対象とした形態素解析辞書の開発，言語処理学会第 14 回年次大会発表論文集，pp.225–228 (2008).
[10] 小木曽智信，小椋秀樹，田中牧郎ほか：中古和文を対象とした形態素解析辞書の開発，情報処理学会研究報告，Vol.2010-CH-85, No.4, pp.1–8 (2010).
[11] 国立国語研究所（編）：太陽コーパス，国立国語研究所資料集 15，博文館新社 (2005).
[12] 近藤明日子：濁点文字使用率から見る濁音表記，雑誌「太陽」による確立期現代語の研究—「太陽コーパス」研究論文集，国立国語研究所報告 122，pp.331–350, 博文館新社 (2002).
[13] 近藤明日子，小木曽智信，須永哲矢ほか：『明六雑誌コーパス』の開発—近代語コーパスのモデルとして，第 2 回コーパス日本語学ワークショップ予稿集，pp.329–334 (2012).
[14] 田中牧郎：言語資料としての雑誌『太陽』の考察と『太陽コーパス』の設計，雑誌「太陽」による確立期現代語の研究—「太陽コーパス」研究論文集，国立国語研究所報告 122，pp.1–48, 博文館新社 (2002).

付　　録

A.1 明六雑誌コーパスの清濁曖昧文字の統計データ

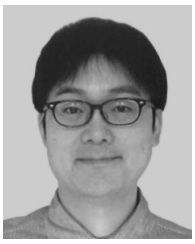
明六雑誌コーパス（2011 年 12 月段階のデータ）内に存在する清濁曖昧文字の統計データを**表 A·1** に示す．明六雑誌の原文はほぼすべての記事が漢字片仮名交じり文で書かれているため，コーパス中の片仮名文字をすべて平仮名に置換したうえで文字のカウントを行った．

表 A·1 明六雑誌コーパスの清濁曖昧文字の統計データ
Table A·1 The statistics of mark-less characters in Meiroku Zasshi corpus.

濁点付与前の表記	出現数	濁点付与後の表記	出現数	例（濁点付与前の表記）
か	1,617	か	1,177	僅かに，自から，明か，恰かも
		が	440	我か，生まれなから，於けるか如く
き	1,766	き	1,712	可き，如き，遠き
		ぎ	54	塞き，過き，防き，注き
く	2,252	く	2,182	全く，能く，曰く，甚しく
		ぐ	70	嗣く，防く，凌く
け	493	け	445	受けたる，設け，傾けしむ，助け
		げ	48	掲けて，妨け，擧けたる，告け
こ	324	こ	322	これ，この
		ご	2	近ころ
さ	1,096	さ	285	起さしめ，示さん，花もさく，驚かさん
		ざ	811	可らさる，根さし，あらさる
し	7,090	し	6,798	配當して，而して，決して
		じ	292	演して，辛うして，禁し
す	5,870	す	4,383	欲する，想像する，失する
		ず	1,487	然らす，可らす，禁する
せ	1,382	せ	1,261	主張せし，調整せん，合せて
		ぜ	121	論せむ，生せん
そ	441	そ	341	凡そ，それ，その
		ぞ	100	何そ，焉そ
た	1,398	た	1,234	得たり，公然たる，除きたり
		だ	164	未た，甚た
ち	475	ち	469	即ち，立ち，克ち，直ちに
		ぢ	6	汝ち，耻ち，攀ち
つ	473	つ	378	二つ，且つ，保つ，曲つて
		づ	95	先つ，一通つゝ，基つきたる
て	6,893	て	6,785	於て，而て，以て，概して
		で	108	好んて，積んて，すてに，今日まて
と	4,740	と	4,148	風俗と慣習と，得たりと，官吏となりて
		ど	592	雖とも，然れとも
は	5,607	は	4,672	婚姻は，有するは，第一には
		ば	935	あらは，非れは，譬へは
ひ	694	ひ	529	悔ひ，救ひ，報ひ，失ひ
		び	165	及ひ，並ひ，一たひ
ふ	1,472	ふ	1,410	思ふ，買ふ，用ふる，言ふ
		ぶ	62	及ふ，學ふ，喜ふ
へ	1,078	へ	491	謂へらく，玉へる，ゆへ
		べ	587	爲すへし，及へり，取調へ
ほ	46	ほ	35	尚ほ
		ぼ	11	亡ほす，及ほす，畧ほ
ゝ	276	ゝ	209	恐るゝ，こゝに，すゝみゆく
		ゞ	67	許さゝる，言はゝ，たゝ
／＼	50	／＼	50	つら／＼，ゆる／＼と，愈／＼，悉／＼く
		／゛＼	0	-

岡 照晃

2010 年豊橋技術科学大学情報工学課程卒業．2012 年奈良先端科学技術大学院大学情報科学研究科博士前期課程修了．現在，奈良先端科学技術大学院大学情報科学研究科博士後期課程在学中．専門は自然言語処理．情報処理学会第 201 回自然言語処理研究会学生奨励賞受賞．



小町 守 （正会員）

2005 年東京大学教養学部基礎科学科科学史・科学哲学分科卒業．2007 年奈良先端科学技術大学院大学情報科学研究科博士前期課程修了．2008 年より日本学術振興会特別研究員（DC2）を経て，2010 年博士後期課程修了．博士（工学）．現在，同研究科助教．大規模なコーパスを用いた意味解析および統計的自然言語処理に関心がある．人工知能学会，言語処理学会，ACL 各会員．

小木曽 智信 （正会員）

1995 年東京大学文学部日本語日本文学（国語学）専修課程卒業．1997 年東京大学大学院人文社会系研究科日本文化研究専攻修士課程修了．2001 年同博士課程中途退学．修士（文学）．2001 年より明海大学講師．2006 年より独立行政法人国立国語研究所研究員を経て，2009 年より人間文化研究機構国立国語研究所准教授，現在に至る．現在，社会人学生として奈良先端科学技術大学院大学情報科学研究科博士後期課程に在学中．専門は日本語学・自然言語処理．日本語学会，言語処理学会各会員．2010 年度山下記念研究賞受賞（フロンティア領域）．

松本 裕治 （フェロー）

1977 年京都大学工学部情報工学科卒業．1979 年京都大学大学院工学研究科修士課程情報工学専攻修了．同年電子技術総合研究所入所．1984～1985 年英国インペリアルカレッジ客員研究員．1985～1987 年（財）新世代コンピュータ技術開発機構に出向．京都大学助教授を経て，1993 年より奈良先端科学技術大学院大学教授，現在に至る．工学博士．専門は自然言語処理．人工知能学会，言語処理学会，認知科学会，AAAI，ACL，ACM 各会員．ACL Fellow．