クラウドソーシングによる形態論情報付与付き辞書整備

岡 照晃 (国立国語研究所コーパス開発センター)

Crowdsourced Compilation of Lexicon with Morphological Information

Teruaki Oka (Center for Corpus Development,
National Institute for Japanese Language and Linguistics)

要旨

本稿では，計算機データベース上の形態論情報付き辞書へ，クラウドソーシングを使い，新情報を大規模に追加する試みを紹介する。当該の辞書は 20 万語規模もの見出し語を有しているが（活用展開や異語形まで含めると 70 万語規模），ここへの新情報追加さえ，例えば見出し語間の関係として「複合語となっている見出し語へ，それを構成している別の見出し語たちはどれか？」といった情報の追加さえ，クラウドソーシングはわずか数日で可能にする。反面，ウェブ上の作業者の大多数は辞書整備に携わったことがなく，中には作業意図を無視する悪質なユーザも存在する。ここでは，こうした作業者たちをどのようにコントロールし辞書整備を行なっているか，具体例を上げて解説する。

1. はじめに

国立国語研究所（以下，国語研）では，現代日本語書き言葉均衡コーパス（BCCWJ）をはじめ，さまざまなコーパス構築が行われている [Maekawa et al., 2014] [Maekawa et al., 2000] [近藤, 2012] [Asahara et al., 2014]。国語研で整備されているコーパスの多くは，形態論情報付きコーパスであり，短単位 [近藤, 2015] と呼ばれる独自に設定した言語単位に分かち書きされ，各短単位に品詞や活用，発音やアクセントといった情報が付与されている。コーパスの構築時に重要な点の一つとして，アノテーションの斉一性の確保がある。分かち書きの粒度をそろえるだけでなく，各短単位に付与されている形態論情報もコーパス全体を通して一貫していない（e.g., コーパス中の異なる位置に出現した同一の短単位に対して，活用など，一部の情報を異なって付与されている）と，ユーザが実際に検索用途でコーパスを利用する際など，検索漏れが生じ得る。そのため国語研では短単位の一覧をデータベース上で一元管理する仕組みを用いてアノテーションの統制を行なっている。このデータベースが電子化辞書 UniDic（UniDicDB）である [伝ら, 2007]。UniDicDB は，所内のコーパスデータベースと参照関係にあり，コーパスデータベース中の短単位は，

- UniDicDB に登録されており，
- UniDicDB 中の一意のエントリを参照する（リンク付けられている）状態になっている（図 1）。

こうしたコーパスと辞書を統合したシステム運営の利点として，先に挙げた斉一性の確保だけでなく，現時点の UniDicDB に存在しない情報（項目）が，新たに UniDicDB へ追加されると，その情報がデータベース間のリンクでコーパス全体へ瞬時に反映（新項目の追加）できる。

例えば，前述したコーパスの中には「走り抜ける」といった複合語も一つの短単位として存在している。しかしこれが「走る」と「抜ける」という 2 つの要素から構成されているといった情報は持っていないため，単純に「走る」でコーパスを短単位検索しても「走り抜ける」は検索から漏れてしまう。そこで，コーパス中の複合語すべてに「走り抜ける」が「走る」と「抜ける」の複合であるといった構成要素の情報を付与したいと考える。このとき，コーパス全体を読んでこの一つずつ新情報を付与する必要

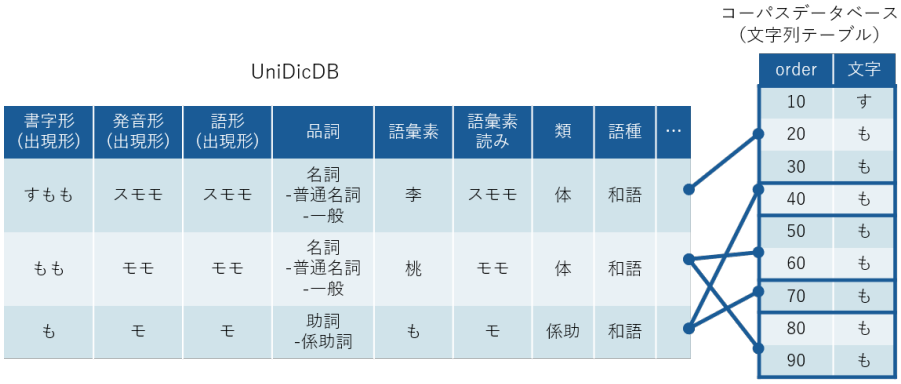

図 1　UniDicDB とコーパスデータベースのリンク関係。コーパス中の同一短単位は UniDicDB 中の一いのエントリを参照する状態となっている。

はない。UniDicDB の中のエントリ「走り抜ける」が，別エントリ「走る」と「抜ける」の複合であるといった情報を付与するだけで，前述したリンク関係を通して，コーパスデータベース側の全「走り抜ける」にこの構成要素の情報がすぐに反映される。このようにコーパス中の全短単位を一つずつ確認していかずとも，UniDicDB のエントリ通して効率的なに新情報の付与をコーパスに対して実施できる。

しかしながら，コーパス全体を見る必要がないとはいえ，UniDicDB には約 20 万件の見出し語（語彙素）が登録されており，各見出し語以下の活用展開，異語形，異表記まで含めると約 70 万件もの短単位を参照する必要がある。この作業を一人ないしは数人で行うにしても早くとも数週間，数か月の日数を要する。

そこで本稿では，クラウドソーシングを使った UniDicDB への新情報追加方法について紹介する。クラウドソーシングは，ウェブ上の不特定多数の作業者たち（**ワーカー**）に，なにかしら作業を分散・並行して依頼できる仕組みであり，上述したような作業も数百〜千人の規模で，数日のうちに完了することができる。反面，ワーカーの大多数は辞書整備に携わったことがなく，中には作業意図を無視する悪質なワーカー（**バッドワーカー**）も存在する。こうしたワーカーたちをどのようにコントロールし，辞書整備を行なうか，実際に行なった複合語への構成要素情報付与を例に上げて，解説する。

2.　クラウドソーシングタスクの設計とバッドワーカー対策

本稿では，実際にクラウドソーシングの仲介サービスを提供している Yahoo!クラウドソーシング[(1)]を例に解説を行なっていく。Yahoo!クラウドソーシングでは，単純なデータチェックやアンケートをあらかじめ用意された約 150 種のテンプレート（図 2[(2)]）から選択して不特定の Yahoo!ユーザへと依頼できるサービスを提供しており，本稿でもそのサービスを利用する。依頼する 1 つの作業（**タスク**）につき，選べるテンプレートは 1 種のみであり，テンプレート 1 枚を 1 **設問**とカウントする。テンプレートによっては 1 設問中に複数の問いを埋め込むことが可能であるが，本稿では 1 設問につき一つの問いを含むテンプレートを使用する（図 2 の No. 18）。ワーカーには一度に複数の設問を提示することができ，今回は一度に 10 設問を 1 セットとし，それを最大 3 回まで行える設定でタスク依頼を行なった。

先述の通り，ワーカーの大多数は辞書整備に携わったことがなく，言語学的な知識も有していない。そんなワーカーたちに「複合語を判別し，構成する要素を選択せよ」という問いを投げても，そもそも「複合語」が何か，彼らにはわからない場合がある。そのため設問の設計は，作業してもらいたい内容

(1) https://crowdsourcing.yahoo.co.jp/
(2) `https://req-crowdsourcing.yahoo.co.jp/requester/request/register/select/template`

図 2　Yahoo!クラウドソーシングで選択可能なテンプレートの一部。

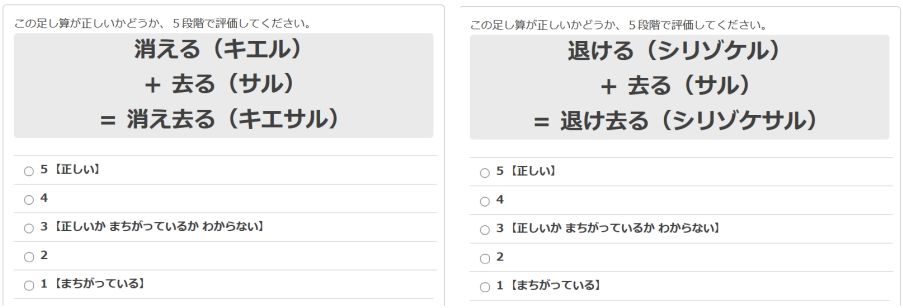

図 3　設問の例。左があらかじめ回答の決まっているチェック設問。右が実際にワーカーに判断をあおぐ設問。

を，専門的な言葉を使わず，より単純にかみ砕いた形で提示する必要がある。そこで今回は，当該短単位と，それをあらかじめ形態素解析器 MeCab[Kudo et al., 2004] と解析用 UniDic(3)を使ってより細かい長さに短単位自動解析した結果を並べ，図 3 のような「単語の足し算が正しいか否か？」という問いとしてワーカーに提示した。5 段階評価にしたのは，ワーカーの多くが選択肢 5 や 1 の【正しい】【まちがっている】という強い判断の表現の選択肢を選びたがらず，【わからない】を選びたがる傾向を考慮したものであり，実際には，集計の際，選択肢 4 と 5，1 と 2 は合わせてカウントする。また細々とした作業解説を書いても基本的にワーカーたちはそれを読まないため，ワーカーには何が【正しい】で何が【まちがっている】のか図 4 のように例示することも重要である。「複合語の構成要素」から「単語の足し算」という，タスクの簡便化を行うと，図 4 の「走る + 猿=走り去る」という例を読みが「ハシリサル」であるため「正しい」と判断するワーカーも現れ得る。また「走る + 去る=走り去る」では，最初に提示されているのが「走り」ではなく「走る」であるため足し算しても「走り去る」にはならない，という考えを持つワーカーもいる。例示はワーカーの作業指針であり，少ない例示でどれだけこちらの意図を伝えるかを設計することはクラウドソーシングを行うにあたって，非常に重要な作業である。

以上のように，ワーカーの知識レベルに合わせたタスク設計を行うことで，言語学的な知識を持たない非専門家であっても，こちらの意図した複合語判別のタスクが可能になる。しかし，いくら我々側が手を尽くしても，タスクの意図を理解しない／する気がない／できない／したつもりになっているワー

(3) https://unidic.ninjal.ac.jp/

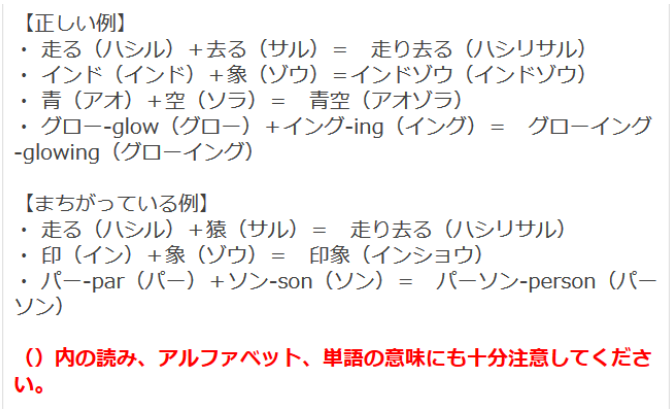

図 4　【正しい】場合，【まちがっている】場合の例示と作業上の注意。

カーは多数出現する。そういったバッドワーカーを除外するため，設問のセットの中には，あらかじめ回答の決まっている**チェック設問**を含めることができる。図 3 の左側はチェック設問であるが，こうした実際に解いてもらいたいタスクと同等か，それよりもさらにわかりやすい問題をあらかじめ大量に回答付きで用意し，その正答率でバッドワーカーの除外を実施する。バッドワーカーの傾向は基本的に以下のタイプに分別できるが，「この程度の理解力は欲しい」という最低ラインを定めてチェック設問を作り，1 タスクを複数回に分け，順番に実施することで，段階的にバッドワーカーを抽出・除外していくことができる。

- タスク開始前の解説文を読まずに，タスクを開始する。
- タスクを行うにあたっての判断力が足りないにも関わらず，タスクを実施する。
- 選択問題の場合，選択肢を読まずに適当に選ぶ。自由記述の場合「あああああああああ」といった意味のない入力を行う。
- 複数の Yahoo!アカウントとテンプレートに合わせた自動化プログラムを作成し，直上のような回答を大量に行う。

またバッドワーカーは土日祝日といった学校や仕事が休みの日，もしくは深夜帯といった空き時間に多くなる傾向があるため，タスクはなるべく平日の朝から夕方の間に開始・完了することが望ましい。バッドワーカーの中にはゲーム間隔でクラウドソーシングを行っている者たちもおり，ネットの匿名掲示板で月の稼ぎやタスクのチェック設問のバラし合いを行なっているため，掲示板を見張り，彼らが除外できているかどうかでチェック設問を調整することも重要である。

チェック設問以外にバッドワーカーの影響を減らし，タスク結果の精度を上げる方法として，同じ設問を複数のワーカーに提示して，その多数決をとるという方法もある。単純な方法であるが効果的であり，それをベイズ統計の手法に発展させ，当該設問の正解確率を求めることや，ワーカーの作業レベル，設問の難度を数値化する手法も盛んに研究されている [鹿島ら, 2016]。

3.　クラウドソーシング実施結果

UniDicDB に含まれる現代語短単位のうち，以下のものを取り除いた全 94,076 語彙素（辞書における見出し語に相当するもの）から，MeCab の自動解析の確信度が高い順に並べ，上位 3 万件（=設問数 3 万）を対象に，1 万件ずつ段階的なタスク実行を行なった。

- 1 文字のみからなる短単位
- 英数字のみからなる短単位

表 1　タスク実施ワーカー中におけるバッドワーカーの割合。

タスク番号	バッドワーカー数/タスク実施ワーカー数
0（1 日目）	336/676 (49.7%)
1（2 日目）	148/792 (18.7%)
2（3 日目）	194/893 (21.7%)
3（4 日目）	123/751 (16.4%)

表 2　タスク 0 実施直後，バッドワーカーを取り除かないまま，同じ設問に取り組んだワーカー 3 人中 3 人が【正しい】を選択した例。

易（エキ）	＋	学（ガク）	＝	易学（エキガク）
黒（コク）	＋	炭（スミ）	＝	黒炭（コクタン）
賢（ケン）	＋	明（メイ）	＝	賢明（ケンメイ）
月（ゲツ）	＋	齢（レイ）	＝	月齢（ゲツレイ）
万（マン）	＋	年（ネン）	＝	万年（マンネン）
東（トウ）	＋	電（デン）	＝	東電（トウデン）
東（トウ）	＋	芝（シバ）	＝	東芝（トウシバ）
仏（ブツ）	＋	教（キョウ）	＝	仏教（ブッキョウ）
星（セイ）	＋	人（ヒト）	＝	星人（セイジン）
旗（キ）	＋	下（シタ）	＝	旗下（キカ）
思い出（オモイデ）	＋	出す（ダス）	＝	思い出だす（オモイイダス）
バックス-backs（バックス）	＋	トップ-top（トップ）	＝	バックストップ-backstop（バックストップ）
賢（ケン）	＋	人（ヒト）	＝	賢人（ケンジン）
主（シュ）	＋	君（キミ）	＝	主君（シュクン）
雄（ユウ）	＋	志（ココロザシ）	＝	雄志（ユウシ）

- 記号，固有名詞

表 1 にタスクに参加したワーカー数と，チェック設問の正答率 70% 以下をバッドワーカーと見なし，段階的に除外していった結果を示す。タスク番号 0 は，試験的に設定した動作検証用タスクであり，3 万件からランダムに選んだ 6,000 件を使用している。この割合を見ればわかる通り，試験タスクの段階で 50% 近く存在していたバッドワーカーを除外することに成功し，以降のタスクではバッドワーカーの割合が 20% 前後まで減少，最終的には 16% まで減少している。

またタスク 0 とタスク 3 の各時点で得られた UniDicDB 中の短単位に分割可能な複合語（3 人中 3 人が【正しい】を選んだ設問）のリストを表 2 と表 3 に示す。表を見比べると，タスク 0 実施直後にワーカー 3 名が【正しい】を選んでいるのは，単純に 2 字の漢語を読みで分解して正しくみえる例であるが，バッドワーカーを除外していくことで，活用変化や連濁も考慮し，こちらの意図を理解した上でタスクに取り組むワーカーに絞り込むことに成功している。また MeCab の短単位自動解析は必ずしも正しい結果を返すわけではなく，タスク 0 では読みや分割を間違った結果も【正しい】と判断しているが，バッドワーカーの排除によってこのエラーも，「焦げ」のように名詞にするか，「隠す」のように動詞にするかといった人によって判断が揺れる段階まで質を上げることに成功している。

4. おわりに

本稿では，Yahoo!クラウドソーシングサービスを利用して，電子化辞書 UniDic に新情報を大規模かつ高速に付与する方法について解説した。ネット上のワーカーたちは辞書整備に携わったこともない非専門家たちであるが，依頼したい作業の形を変え，タスク設計を工夫することで，非専門家でも判断が行えるアノテーションタスクとして依頼が可能となる。またネット上には，タスクの意図を理解しない／する気がない／できない／したつもりになっているバッドワーカーが多数存在するが，チェック設問と多数決をうまく利用することで，彼らを除外し，より精度の高いアノテーションを実現できることを

表 3　タスク 3 実施後，同じ設問に取り組んだワーカー 3 人中 3 人が【正しい】を選択した例。

夜（ヨル）	＋	桜（サクラ）	＝	夜桜（ヨザクラ）
黒（クロ）	＋	焦げ（コゲ）	＝	黒焦げ（クロコゲ）
隠す（カクス）	＋	戸（ト）	＝	隠し戸（カクシド）
三（サン）	＋	塁（ルイ）	＝	三塁（サンルイ）
取る（トル）	＋	置く（オク）	＝	取り置く（トリオク）
雪（ユキ）	＋	雲（クモ）	＝	雪雲（ユキグモ）
ポスト-post（ポスト）	＋	カード-card（カード）	＝	ポストカード-postcard（ポストカード）
白（シロ）	＋	薔薇（バラ）	＝	白薔薇（シロバラ）
初（ハツ）	＋	泳ぎ（オヨギ）	＝	初泳ぎ（ハツオヨギ）
飾り（カザリ）	＋	結び（ムスビ）	＝	飾り結び（カザリムスビ）
編み（アミ）	＋	棒（ボウ）	＝	編み棒（アミボウ）
白（ハク）	＋	金（キン）	＝	白金（ハッキン）
括り（ククリ）	＋	染め（ソメ）	＝	括り染め（ククリゾメ）
ブック-book（ブック）	＋	エンド-end（エンド）	＝	ブックエンド-book end（ブックエンド）
茜（アカネ）	＋	空（ソラ）	＝	茜空（アカネゾラ）

示した。

ただしこのような手法で作成したデータは，全体が必ずしも正しいというわけではなく，自動解析時のエラーが原因となる取りこぼしや，非専門家ゆえの誤った知識が混入する恐れがある。そのため，作成したデータをそのまま国語研の UniDicDB へ取り込むことはできない。最終的には専門家による全件確認か，もしくはあくまでクラウドソーシングで作成した，エラーを含む可能性のあるデータということを断った上でユーザに別途配布する形式がふさわしいと考えられる。実際すでに，同様に作成したデータを UniDic 非コアデータとして，UniDicDB 中の語彙素 ID（[小木曽ら, 2014] 参照）に紐づける形で公開しており(4)，今後も様々なデータを構築・追加していく予定である。

謝　辞

本研究は国立国語研究所の所長裁量経費の助成を受けたものです。

文　献

[Asahara et al., 2014]　Asahara, M., Maekawa, K., Imada, M., Kato, S., and Konishi, H. (2014). Archiving and Analysing Techniques of the Ultra-large-scale Web-based Corpus Project of NINJAL, Japan *Alexandria*, 25(1-2), pp.129–148.

[Kudo et al., 2004]　Kudo, T., Yamamoto, K. and Matsumoto, Y. (2004). Applying Conditional Random Fields to Japanese Morphological Analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*(*EMNLP-2004*), pp.230–237.

[Maekawa et al., 2000]　Maekawa, K., Koiso, H., Furui, S., and Isahara, H. (2000). Spontaneous speech corpus of Japanese. *Proceedings of the Second International Conference on Language Resources and Evaluation* (*LREC'00*).

[Maekawa et al., 2014]　Maekawa, K., Yamazaki, M., Ogiso, T., Maruyama, T., Ogura, H., Kashino, W., Koiso, H., Yamaguchi, M., Tanaka, M., and Den, Y. (2014). Balanced corpus of contemporary written Japanese *Language Resources and Evaluation*, 48, pp.345–371.

[小木曽ら, 2014]　小木曽智信，中村壮範 (2014).『現代日本語書き言葉均衡コーパス』形態論情報アノテーションシステムの設計・実装・運用自然言語処理，21, 2, pp. 301–332.

[鹿島ら, 2016]　鹿島久嗣，小山聡，馬場雪乃 (2016). ヒューマンコンピュテーションとクラウドソーシング，機械学習プロフェッショナルシリーズ，講談社.

[近藤, 2012]　近藤泰弘 (2012).「日本語通時コーパスの設計」NINJAL「通時コーパス」プロジェクト・Oxford VSARPS プロジェクト合同シンポジウム「通時コーパスと日本語史研究」予稿集，pp.1–10.

[近藤, 2015]　近藤泰弘 (2015).「『日本語歴史コーパス』と日本語史研究」コーパスと日本語史研究，ひつじ研究叢書＜言語編＞，第 127 巻，ひつじ書房，pp.1–16.

[伝ら, 2007]　伝康晴，小木曽智信，小椋秀樹，山田篤，峯松信明，内元 清貴，小磯 花絵 (2007).「コーパス日本語学のための言語資源：形態素解析用電子化辞書の開発とその応用」『日本語科学』，22 号，pp.101–123.

(4) https://github.com/teru-oka-1933/unidic_non_core