2022 年度 修士論文

曖昧性に着目した
読みづらさの検出

指導教員　　河原 大輔 教授

研究指導名　　自然言語処理研究

早稲田大学 基幹理工学研究科 情報理工・情報通信専攻

学籍番号 5121F093

吉田 あいり

2023 年 1 月 23 日

概要

読みやすい文章は表現や伝達の補助の面で執筆者にとって有益であり, 理解や消費時間の面において読者の利益となる. 執筆支援として読点挿入や語順整序が行われているが, その前段階として読みづらさの検出を提案する. 読みづらさの評価の基準としては構造的曖昧性と語義曖昧性を活用する. 構造的曖昧性とは文の構造解釈に複数の選択肢があることで, 語義曖昧性は意味曖昧性の 1 つであり, 対象単語における語義の判別ができないものである. 構造的曖昧性を活用する手法では, 先行文脈の情報が多いほど読みやすいことや誤読が読み負荷を高めることに着目する. 音声処理で用いられることの多い漸進的係り受け解析をテキストに適用することで人間が文を読む状態を再現し, 係り受け構造の保留を使用した検出手法を提案する. 語義曖昧性に関してはまず日本語の WiC データセットの作成及び拡張を行い, このデータを活用して対象単語の複数語義における WiC の結果を用いて読みづらさの検出を行う. WiC とは 2 文に含まれる同じ単語の語義が一致するかを判定するタスクである. どちらの曖昧性を用いた検出においても一定の精度の検出を行うことができた. 今後の課題は, 対応単語の更なる拡大や読みづらさの検出におけるグレードの導入を行うことでより適切な支援を行うことである.

目 次

図 目 次

表 目 次

1 はじめに

テキストを読む機会は多く, 文章が読みやすいことは消費される時間の短縮以外に読者が文章を誤解することを防ぐ. また, 読みやすい作文ができているということは執筆者の主張の要点も伝わりやすい. そのため, ここで求められる読みやすい文章とは, 情報を損なわないが簡潔であり誤読しづらいものである. 自動的な執筆支援の研究はなされており, 実際に行われているものとしては, 読点の挿入 [1, 2] や語順整理 [3, 4] 等が挙げられる. 構文構造に着目し, 関係を保ったままの並び替えや挿入が行われる. これらの支援では実際に文章に対して能動的な加工が行われているが, その前段階の支援として読みづらさ自体を執筆者に提示するフェーズの必要性があると考える. 単に自動的に文章の改善を行うのみでは執筆者の執筆能力に対する寄与はなく, 同じ誤りを繰り返すことに繋がる. そこで必要となるのが, 実際にどこが読みづらいのかという指摘である. 言葉を用いて表現したり伝達する能力を養うことが教育における課題として挙げられており, 文章を書くことへの苦手意識が強いことが知られている. 作成した文章に対して, どこに起因して読みづらくなっているのかを可視化することは, 改善すべき点が明確になることから苦手意識の払拭ないし作文能力の向上へ繋がると言える. そのため本研究では読みづらい文を検出することを目的とする.

ここで読みづらさとは何であるかを考える. 単語の意味が難解である文ややたらと句切れがなく長い文章は読みづらい. しかし, 単に漠然と読みづらい文というのでは評価ができず, 何らかの基準が必要となる. 現状, 人における読みづらさは読み時間を活用した評価がなされている. どこに注視しているのかなど詳細な分析ができる代わりに扱いは難しく, データの拡張もコストがかかる. 機械においては言語モデルにより計算されるサプライザルを用いて評価されており [5], 前述から予想しづらいことは読み負荷が高まるというサプライザル理論に基づく [6, 7]. 本研究ではまず執筆者に対して個別事象における読みづらさを認識してもらうことが目的であり, 言語の曖昧性に着目して読みづらさを扱うこととする. 曖昧性には構造的なものと意味的なものがあり, その中でも特に構造的曖昧性と語義曖昧性に焦点を当てる.

構造的曖昧性とは構造解釈に複数の選択肢があることである. 日本語の逐次的な読みやすさにおける先行研究にて, 先行文脈に係り元が多い文節ほど読みやすいことや読み間違いが読み負荷を高めることがわかっている. 例えば, 文 (1) [8] は「男性を」の部分で再解釈に要する読み時間が増加する.

(1) 警官が犯人を捕まえた男性を...

構造的曖昧性に着目した手法では, 漸進的構文解析結果と係り先の保留のプロセスを利用して読みづらさの検出を行う.

意味的曖昧性の一つには語義曖昧性がある. 例えば「動きを踏まえた分析」という文が与えられた時に「動き」が傾向を表すのか実際の活動を表すのかの判別が付かないというようなものである. 語義曖昧性に着目した手法では, まず, 2 文に含まれる同じ単語の語義が一致するかを判定する WiC (Word in Context) データセット [9] の日本語版である JWiC を構築する. JWiC は日本語フレームネットから構築するが, それだけではデータ数が少ないため, データ拡張を行う. 作成した JWiC を利用し, 対象単語が持つ語義と WiC を行うことで語義曖昧性判定に基づいた読みづらさの検出を行う.

本論文での章構成は以下の通りである．2 章ではまず読みづらさの分析や曖昧性の関連研究について述べる．3 章では構造的曖昧性に基づいた読みづらさの検出について述べる．4 章

で JWiC の構築について説明したのち，5 章で語義曖昧性に基づいた読みづらさの検出を行う．6 章にて実験結果や提案手法についての議論を行い，7 章でまとめを行う．

2 関連研究

本研究で扱う読みづらさにおける先行研究について述べる. 続いて, 検出に使用する構造的曖昧性と語義曖昧性に関連する研究内容について述べる．また, 使用するデータセットの特徴についても記載する. 曖昧性の解消事例よりも実際に扱う曖昧性に着目する.

2.1 読みづらさの分析

人間の読みづらさは読み時間を用いて評価されており, 読み時間を主軸とした分析がされる. 日本語の読み時間データは BCCWJ-EyeTrack が整備され, 先行文脈に係り元文脈が多い要素ほど読み時間が短くなるということが分析されている [10]. また, 日本語の読解時間と統語・意味カテゴリの対比分析においても関連する先行文脈が読み時間の減少に関与すること [11] や, 文の意味的曖昧性の高さが構造的曖昧性の解消/保留に影響することがわかっている [8, 12]. 他にも個別の言語現象が読み時間に与える影響に対して分析が行われている [13, 14]. 計算機における読みづらさはサプライザルで評価され, これは先行文脈からの単語の尤度であり確率の低い単語が続いた場合その文は読みづらいということになる. [5]. 予想と異なる内容が現れた際に読み負荷が高まるというサプライザル理論に基づいている [6, 7].

これらは読みやすい文における知見ではあるが, 統一的な傾向は定かではない. そこでサプライザル理論に基づいて, 日本語読み時間に関する傾向がサプライザルが大きいところで読みにくくなるという傾向に統一的に解釈できるかという仮説検証がなされた [15]. 読み時間の様々な傾向がサプライザルでも再現され, 情報量の観点から解釈できることが示されている.

曖昧性に着目した読みづらさの検出を行うが, 複数の曖昧性があり以下のようなものが挙げられる.

構造的曖昧性 「警察が犯人を捕まえた男性を...」

語義曖昧性 ドライバー → 運転手, 工具

省略・照応 (共参照) 席替えで山本さんと隣になった。彼女は…

談話関係 雨が降ったから、水溜りができるはずだ。(根拠)
危ないから、雨の日に川に近づいてはいけない。(原因・理由)

本研究ではこの中から特に構造的曖昧性と語義曖昧性に着目し, 読みづらさの検出手法を提案する.

表 1: 単語親密度の例

見出し語	読み	KNOW	WRITE	READ	SPEAK	LISTEN
這般	しゃはん	-1.30	-1.58	-1.63	-1.48	-1.57
既存	きぞん	1.99	0.29	1.02	0.10	0.42
実状	じつじょう	1.02	-0.18	0.05	-0.15	0.05

2.1.1　単語親密度

本研究では構造要素による読みづらさに着目する際や使用単語の選別において, 単語難易度による影響を減らす工夫が必要である. 単語に対するの評価指標の 1 つに単語親密度があり, これはその単語がどれくらい使用及び知られているかを表している.

実際に構築されているデータベースとしては WLSP-familiarity[1]がある. これは日本語のシソーラスである『分類語彙表増補改訂版』 [16] の電子化データ『分類語彙表増補改訂版データベース』の語彙項目を対象に親密度情報を付与したもの [17] であり, これを用いてフィルタリングを行うことで見知らぬ単語による読みづらさを軽減することができる. このデータベースにおける単語親密度は知っている　(KNOW) かどうかだけでなく, 書く (WRITE)・読む (READ), 話す (SPEAK)・聞く (LISTEN) の項目も含み, 複合項目も存在する. 項目ごとの分布は KNOW が高い親密度の傾向がみられ, また, 受容過程の READ と LISTEN が高く，生産過程の WRITE と SPEAK は低い傾向がみられる. 値は概ね-2 から 2 に分布している．表 1 に実際の語彙と単語親密度を示す.

2.1.2　ガーデンパス効果

読み時間が増加する要素の 1 つとしてガーデンパス (GP) 効果がある. これは, 文の理解の途中で一時的に誤った解釈をし, 続きを読んだ際に誤解に気づいて解釈をし直す事になるが, この再解釈により発生する処理負荷や読み直しにかかるコストにおける効果である. 文理解の初期段階で曖昧性を解消しようとするために, 結果として誤解釈が発生する. この効果がもたらす構造的曖昧文の性質は言語の構造により異なることが分析されている [18]. 例えば次の例では「怪我をしている犬」を抱き上げている女性の記述として読み進めると,「怪我をしている女性が犬を抱き抱えていたことがわかる.

1. 怪我をした犬を抱いていた女の人がその後救急車で搬送された。

GP 文は最後まで解釈が曖昧な場合もある. 例えば,「かわいい少女の猫」の「かわいい」が修飾するのが「かわいい」なのか「猫」なのかはこの部分のみでは決定できない.

2.2　構造的曖昧性

構造的曖昧性とは複数の解釈ができるために文の解釈が定まらないものである. 次の例文のようなものが挙げられる.

[1] https://github.com/masayu-a/WLSP-familiarity

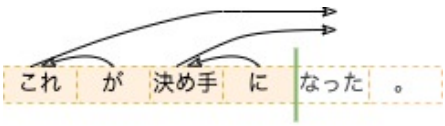


図 1: 未入力範囲への係り受けを許容する漸進的構文解析

1. 私は旅先で道に迷ってそこに立っている女性に声をかけた

この文において, 確定しているのは「私」が「女性」に「声をかけた」である. 道に迷っているのは私なのか女性なのか, 旅をしているのはどちらかなどはここでは確定できないため以下のような解釈ができる. 解釈はこれらに限らないがこの文のみでは判断することができない.

1.1 私が旅先において道に迷い, そこに居合わせた女性に対して声をかけた

1.2 旅に来た女性が道に迷い, 私がその女性に声をかけた

2.2.1　漸進的係り受け解析

人間の言語理解過程には漸進性があり, 音声言語アプリケーションの基盤技術として漸進的係り受け解析技術の研究がなされている [19]. 言語処理過程のデータを構築・分析することにより人間の漸進的係り受け解析能力や入力予測能力の解析がなされている [20, 21]. 逐次処理を行うために, 係り先を未入力部分と見做し, 未入力の中でも同一文節であるのかまで考慮してアノテーションが行われている. 係り先文節を考慮しない場合の, 未入力範囲への係り受けを許容する構文解析を図 1 に示す. 人間の言語理解には漸進性がある [22] が, 特に音声では入力と同時に処理することが求められ, 漸進的言語処理システムが開発されている. 本研究ではこの漸進性をテキストに組み込むことで, GP 効果等の特徴を考慮した手法を提案する.

2.3　語義曖昧性

語義曖昧性とは対象単語がどの語義で使用されているのかがわからないことであり, ひらがなの表記で起こりやすく「はし」のみでは「橋」, 「箸」, 「端」のいずれであるのか判別できない. ひらがなに限った話ではなく, 文脈が長いほど起こりづらいが短文において起こりやすい.

これを解消するタスクも存在し, 語義曖昧性解消 (WSD) は対象単語である多義語が辞書におけるどの語義で使用されるのかを識別する分類タスクである. WordNet [23] の辞書から抽出されたラベルの意味区分や語義の定義文と用例文を使用した手法が一定の効果を挙げており [24], さらに追加の用例を使用したり定義文と例文を組み合わせた手法も提案されている [25, 26, 27].

2.3.1　WiC

言語理解のためのベンチマークとして SuperGLUE [28] が構築されており, 8 つのタスクのうちの 1 つである語義曖昧性解消タスクが WiC (Word in Context) [9] である. SuperGLUE の

表 2: SuperGLUE の構成

タスク	識別子	内容
CommitmentBank	CB	テキスト含意タスク (NLI)
Choice of Plausible Alternatives	COPA	因果推論タスク (QA)
Multi-Sentence Reading Comprehension	MultiRC	質問応答タスク (QA)
Recognizing Textual Entailment	RTE	テキスト含意タスク (NLI)
Words in Context	**WiC**	語義曖昧性解消タスク (WSD)
The Winograd Schema Challenge	WSC	共参照解決タスク (coref.)
BoolQ	BoolQ	質問応答タスク (QA)
Reading Comprehension with Commonsense Reasoning	ReCoRD	質問応答タスク (QA)

表 3: WiC の例

ラベル	対象	例文 1	例文 2
F	bed	There's a lot of trash on the bed of the river	I keep a glass of water next to my bed when I sleep
F	justify	Justify the margins	The end justifies the means
T	air	Air pollution	Open a window and let in some air

タスク一覧を表 2 に示す. WiC は 2 文に含まれる多義語である同一単語が同じ語義で使用されているかを判断する 2 値分類のタスクであり, 例を表 3 に示す. 「bed」の例において例文 2 では寝具の意味で用いられているが, 例文 1 では「on the bed of the river」で河床の意味で使用されているため異なる語義である. 対して,「air」の例文においてはどちらも空気の意味で使われており語義はどちらも同じである.

多言語 WiC データセットとして XL-WiC [29] が整備されているが, 完全に整備されている言語はドイツ語, フランス語, イタリア語に止まり, その他の言語は dev と test データのみである. 日本語もデータセット内に含まれているが, 1,000 組程度と数が少ないため言語資源として活用するのは困難である. 従って, 日本語データセットの構築が求められているのが現状である.

2.3.2　JFN

日本語フレームネット (JFN) [30] は日本語における語彙・構文複合言語資源であり, 言語形式とその意味の関係を背景知識 (フレーム) との関係で捉えている. 対象語句を含む例文にフレームがアノテーションされている. 例えば「持つ」に関する以下の例文には次のフレームが付与されている. このフレームを語義曖昧性解消のデータとして活用する.

1. … 報告を持ってきた… : Bringing
2. … 在庫の持ち方を変えて… : Storing
3. … 特許権を持っています。 : Possession

3 構造的曖昧性に基づく読みづらさの検出

3.1 BERT に基づく構文解析

構造的に曖昧である文の検出を行う. しかし, 構造的に曖昧であることを明示したデータセットはなく, 曖昧な文とそうでない文を収集及びラベル付けを行うのはコストが非常に大きい上に困難である. そこで本研究では読みづらさを検出するために, 機械学習を用いた構文解析を行い, その結果を活用することでラベル付きデータを使用しない手法を提案する.

BERT [31] を活用し日本語構文解析の精度を向上させる手法 (BERTKNP) が提案されており [32], これは構文解析を head [33] の選択と見なすことにより BERT に 1 層追加する形で実装を行っている. 事前学習されたモデルを活用することで, 大量の生コーパスを利用し構文解析の精度向上を達成している. ここに漸進的な要素を盛り込むことで機械学習を用いた漸進的手法が実現できる.

3.2 漸進的構文解析手法

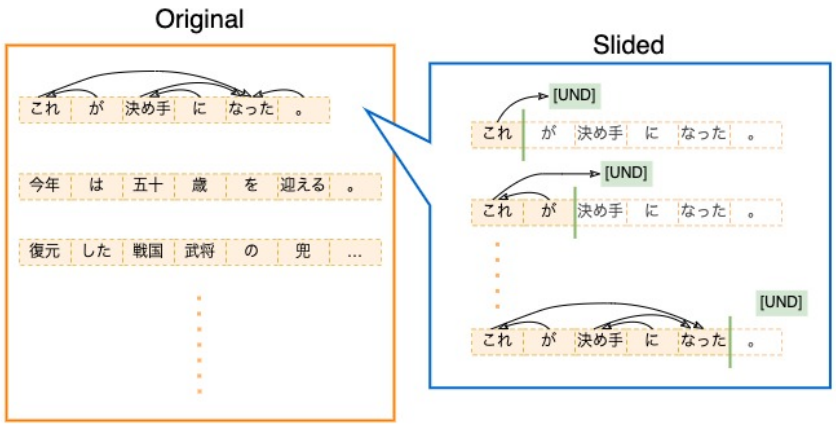
BERT による構文解析に漸進的な視点を入れることで, 人間が読む際の要素を追加する. つまり, 文において単語が前から順に読まれることを重視し, 係り受け構造の学習データを作成する.

単語ごとに係り受けのアノテーションがなされた文 (Original) に対して, 単語が 1 つずつ順に入力される過程 (Slided) を考える. Original においては全ての単語が係り先情報を持っている. これをある単語が入力されるまでの部分までに制限し, 入力される単語数を 1 ずつずらしていくことで漸進的なデータを実現する. 図 2a のように Slided の入力は文の途中までであり, 未入力部分に係る単語も存在する. その場合は係り先を特定単語ではなく, 未決定 ([UND]) と指定する [20] ことで特定の係り先がないことを表現する. 文頭から順にずらしていき, 全てのパターンに対して Slided を作成すると 2b のようになり, Original に対しての分量が非常に多くなり Slided について過学習を起こす可能性がある. そこで Slided には図 2c のように n%の制限を設ける. n%の制限がかけられるのはスライドされている不完全な文のみであり, 完全な文のデータは必ず含まれる. どの程度漸進的データを追加するべきであるかと作成したデータの全体で順番のシャッフルが必要であるのかは予備実験にて確認する.

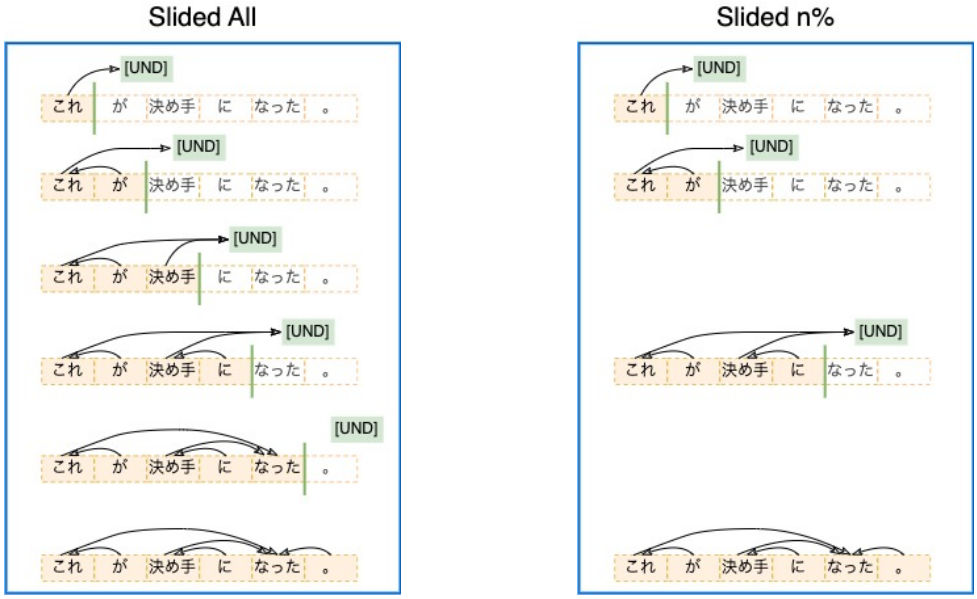
3.3 解析結果

漸進的構文解析において適切な漸進的データ量を確認するために, 実験の結果を表 4 に示す. 精度は単語単位での精度と文単位の精度の平均の 2 通りで算出した. 単語単位においてはどの文に含まれているのかは関係なく全単語の予想係り先の正誤の精度であり, 文単位はある文においての精度を計算しさらにその精度の平均をとっている. スライドする前の文が同じ物でも違う文セットとして扱う. これに加えて [UND] の再現率も測定した. 日本語 BERT は NICT BERT 日本語 Pre-trained モデル[2]の BPE ありのものを使用した. fine-tuning には京都大学テキストコーパス (新聞) と京都大学ウェブ文書リードコーパス (ウェブ) の 2 種類を混合し

[2] https://alaginrc.nict.go.jp/nict-bert/index.html



(a) 漸進的構文構造データ作成



(b) スライドしたデータ　　(c) n%選択したデータ

図 2: 漸進的構文構造データの例

たコーパス約 5 万文を使用し, 元論文にならい 3 epoch 回した. また, 評価には京大コーパス約 4,000 文を使用した.

京大コーパスにおいては加工しない BERTKNP が一番精度が高い. 漸進的要素を加えた Slided All (Orig+Slided 100%) においては元のデータに 5%混合したデータを用いて学習したモデルが一番精度が良く, 再現率は 50%混合した場合が良い値となった. また, 予測精度と再現率はデータの順番をシャッフルすることによる大きな変化は見られず, 無影響ないし悪影響になるため不必要であることがわかった. 漸進的データの必要以上の追加は精度の向上に直結しない上, 計算資源と時間を消費する. 本研究では, 精度を重視して 5%のみ追加したデータで fine-tuning したモデルを用いてこれ以降の実験を進める.

表 4: 漸進的構文解析の精度

モデル	京大 (新聞)	Slided All	UND Recall
BERTKNP	**0.965 / 0.964**	0.760 / 0.852	0.000
Orig+Slided 5%	0.963 / 0.963	**0.958 / 0.956**	0.889
Orig+Slided 20%	0.961 / 0.961	**0.958** / 0.955	0.891
Orig+Slided 50%	0.959 / 0.960	**0.958** / 0.954	**0.892**
Orig+Slided 90%	0.957 / 0.958	0.957 / 0.953	0.890
Slide All	0.958 / 0.958	0.957 / 0.953	0.888
Slide All (shuf)	0.957 / 0.957	0.956 / 0.952	0.887

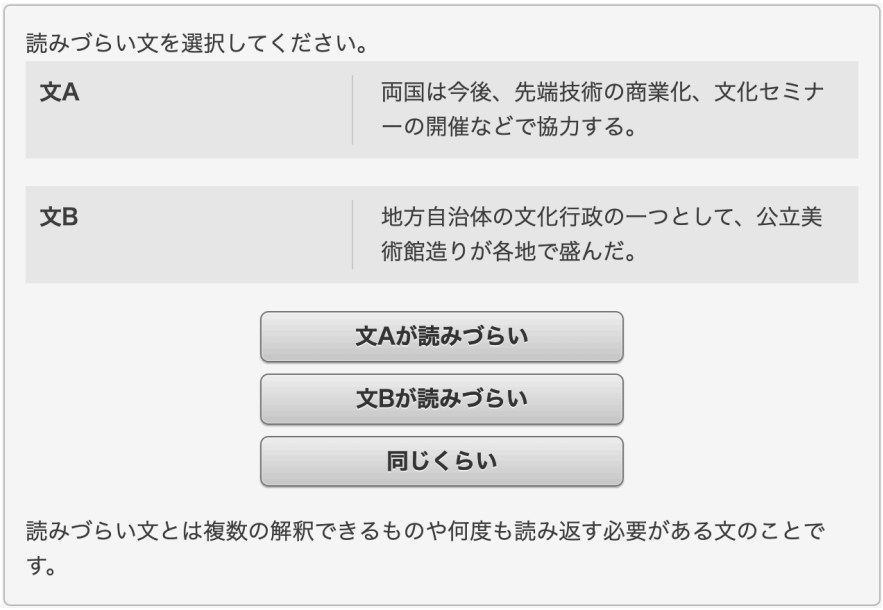


図 3: 構造的読みづらさのクラウドソーシング

3.4　正解データ作成と評価方法

評価に使用する読みづらさの正解データを作成するために, 京大コーパスに前処理として単語親密度によるフィルタリングを行い, 親密度が負のスコアを持つ形態素を含む文を除外した. また, 形態素数が 20 に満たない文も除去し, 形態素数が 20 から 30 のものと 30 以上のものに分割後, 文長によるソートを行ってから文字数順に 2 文のペアを作成した. これにより, 単語難易度による難読性を排除した上で, 文長だけでなく形態素数の近い分ペアを作成している. このペア文の比較による評価を行う. Yahoo!クラウドソーシングにより 1 ペアに対して 10 人の回答をフィルタリングされた 322 文について収集した. 2 文のうちどちらが構造的に読みづらいかという質問に対して, [文 A, 同等, 文 B] の 3 つの選択肢を用意した. 実際の作業画面を図 3 に示す.

文法構造を考慮した読みづらさとして以下の 2 つの要素を挙げ, これに基づき評価させた.

1. 複数の解釈ができる
2. 理解に反復が必要となる

また, 単語の難易度に関しては考慮しないように指示している. 収集データから 2 種類の正

表 5: 各データの文ペア数

データ	n / m	A	B	同等	total
多数決	5	75	100	72	247
	6	42	54	43	139
	7	21	32	18	71
平均	0	131	160	-	291
	1	100	125	-	225
	2	67	91	-	158
	3	52	70	-	122

解データを作成し, それぞれのデータにおいて評価を行う. 作成された正解データのペア数を表 5 に示す.

多数決　10 人中 n 人の回答が一致したペア文を正解データとする. 例えば, [文 A, 同等, 文 B] の回答数が [6, 3, 1] の際は, n=5 の場合は文 A が読みづらいデータとして採用され, n=7 の場合は使用しない. 評価は [文 A, 同等, 文 B] の全ての項目の精度に対して行う.

平均　[文 A, 同等, 文 B] の回答をそれぞれ [-1, 0, 1] の数値と見なして各回答数の和をスコアとする. 閾値 m を使用し, $[-10, -m), [-m, m], (m, 10]$ をそれぞれの回答として分類する. m の値は難易度の判別がつかない文ペアが十分除かれたものを選択する. 正負でスコアリングしているため, [文 A, 同等, 文 B] の回答数が [2, 7, 1] と [5, 1, 4] の場合のスコアはどちらも-1 となる. 従って, この手法では同等である回答が多くとも判別できないため, 文 A ・ B のどちらが読みづらいかのみを扱い, 同等なものは除外した精度により評価する.

多数決・平均手法共に, n/m を大きくするほど確実に読みづらい文ペアを取得できるが, 使用できるペア数が減少するため適当な n/m を抽出して評価を行う.

3.5　読みづらさの検出方法

作成した漸進的データを用いて学習した構文解析モデルによる解析結果から係り先が未決定 ([UND]) である数を長さで正規化したものを読みづらさのスコアとして提案する. 人間の言語理解の漸進性を取り入れた上で, 係り受け構造の保留により受ける読みづらさを考慮している. 係り先の保留が多くなるほど, その文を読む中で構造が決定していない部分が多いということであり, 読みづらいということとなる. 本研究では 2 文のどちらが読みづらいかまたは同等であるかを正解として扱うが, スコアは数値で算出される. そこでスコアの差分を取り, その絶対値が閾値以下のものを同等である場合として評価し, 優位な差がある場合はスコアがより大きい方が読みづらい. 閾値は各 n/m において精度が最も高くなる値を選んだ.

ベースラインには計算機による読みづらさの指標であるサプライザルとして $-\log p(x|$ 先行文脈$)$ を使用する. 実装上では $-$log_softmax を使用した. サプライザルは尤度の反転であり, これが大きいほど読みづらいと言える. これを計算するために言語モデル GPT-2[3] を利用した. こちらもスコアの差分から同等である場合の評価を行った.

[3] https://huggingface.co/colorfulscoop/gpt2-small-ja

表 6: 読みづらさ検出の精度

データ	n / m	ベースライン	提案手法	正解
多数決	5	0.360	**0.368**	0.397
	6	**0.396**	**0.396**	0.453
	7	0.394	**0.451**	0.521
平均	0	0.478	**0.485**	0.519
	1	0.476	**0.493**	0.542
	2	0.456	**0.483**	0.557
	3	0.410	**0.500**	0.549

3.6　結果

表 6 に各データにおける精度の結果を示す. 正解ラベルの結果は漸進的構文解析が全て正しく付与されたデータを用いて提案手法により算出した精度である. 提案手法の結果は構文解析における間違いを含んでいるため, この正解ラベルよりも精度が低下している. 提案手法は多数決手法と平均手法のどちらで作成されたデータに対しても読みづらい文の検出をすることができた. 確実に難易度に差があると判断された文ペア (多数決 n=7, 平均 m=3) に対しては特に有意な差が現れた. 閾値 n/m は大きいほど制限が厳しくなるため, 難易度の差が大きいものの含有率が大きくなる.

4　JWiC の構築

日本語フレームネット (JFN) [30] を用いて日本語版 WiC (JWiC) の構築を行う. オリジナルの WiC は多様性とバランスに注目しており, JWiC においてもこれを重視し収録語彙の汎用性を高めることを目指す. 以下では, JFN の特徴の確認からはじめ, 構築の流れ, モデルによる評価までを述べる.

4.1　日本語フレームネット (JFN) の分析

JFN は日本語における語彙・構文複合言語資源であり, 言語形式とその意味の関係を背景知識 (フレーム) との関係で捉えている. 対象語句を含む例文にフレームがアノテーションされている. 実際の例は 2.3.2 節にある通りである.

収録語彙は 5,000 語程度であり, 名詞が半分以上を占め, 名詞と動詞でほとんどを占めている. 品詞の分布数を表 7 に示す. また, 単語ごとに定義されているフレーム数にも偏りが大きく, 最大で 11 フレームを持つ. 対象単語が持つフレーム数の top 5 を表 8 に示す. 付与されている例文は異なる語彙において共有されており, ユニークな文についての分析が必要である. 図 4 にユニークな文の文長の分布を示す. フレームが付与されている例文の長さは 50 字程度が最も多く, 大抵の例文は 200 字以内の長さをとっている. また,「雇用再生集中支援事業」のような固有名詞が含まれており, 対象が汎用語句に留まらないことが問題であり, 活用にあたってフィルタリングが必要となる.

表 7: JFN の品詞数

POS	n	v	an	a	...	total
num	3302	1109	147	126	...	4913

表 8: Frame 数 (上位 5 件)

Name	品詞	個数
思う	動詞	11
話	名詞	10
言う	動詞	9
中	名詞	6
作る	動詞	6

4.2　JWiC の構築手法

JWiC の構築は, JFN 例文のフィルタリング, 例文のペアリング, 作成したペアのクラウドソーシングによる検証という 3 段階で行う.

JWiC を特定領域に特化したものではなく, 汎用性の高いものとするために対象語句を制限する. 適切な汎用性を保つために, 対象語句の構成単語数とフレーム数に着目する. 本研究では 4 単語以上で構成されるものと, 6 フレーム以上定義されている語句を除くこととした. そのほかにも, 特定の固有名詞や Dirty word, 対象語彙と例文に含まれる対象語彙の漢字が異なるものに関しても排除する. 次に, 対象語句の制限に続いてバランスの調整を行う. 例文は, 文として成立していることと簡潔であることが求められるため, 例文長は 15 字以上 100 字未満とする.

WiC の形式にするために, 対象語句ごとに例文のペアを作成する. ここで, 1 対象語句に対して例文ペア数は 50 ペアまでと制限することで偏りを少なくする. ペアである 2 文が持つフレームが一致するか否かで WiC 用のラベル「同じ」「違う」を付与する. 例を表 9 に示す. 「聞く」の例文においてはそれぞれ尋ねる意味と伝聞の意味で使われているため語義が異なるが, 「見る」の例文はどちらも視覚としての語義で使用されている.

作成したデータに関して, 付与されたラベルの妥当性を確認するためにクラウドソーシングによる検証を行う. ここで, 人手であらかじめ正解ラベルを付与したチェック設問の正答率を確認することでクラウドワーカーの品質を保証する. チェック設問は最も正答率が低いもので 70%を上回るものを使用し, 平均 8 割程度の正答率のものを選択した. 実際に使用した例を表 10 に示す. クラウドワーカーには短文によって例を提示し, チェック設問があることを説明している. 文ペアを提示し, 対象語句の語義が「同じ」か「違う」かを選択してもらうが, 適当な回答をできる限り取り除くために「わからない・判断できない」という選択肢も用意した. 実際の作業画面を図 5 に示す.

クラウドソーシングには Yahoo!クラウドソーシングを使用し, 1 ペアに対して 20 人から回答を回収した. 解答の結果は表 11 の通りで, クラウドワーカーの正答率のばらつきを確認するために四分位数を示す. またこの分布の箱ひげ図を図 6 に示す.

この結果から極端に精度の悪い文ペアを除いて JWiC の完成とする. JFN フレーム付けは細かく, 見分けのつかないものも含まれるため正答率がさほど高くならなかったと考えられる. 3,230 ペアに対して精度が 0.35 以上のものを採用し 2,495 ペアを得た. 含まれる対象語句

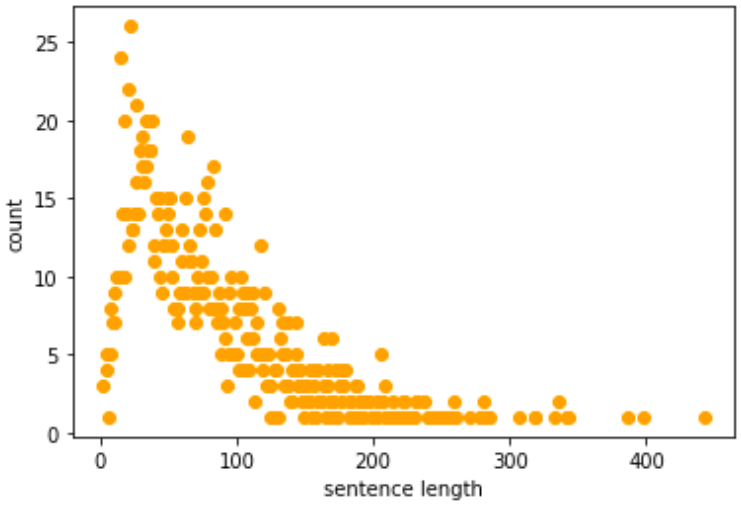


図 4: 文長の分布 (ユニーク)

表 9: JWiC の例

対象	ラベル	例文ペア
聞く	違う	社内で聞いてみても誰もわからない。 奥さんがまだお元気と聞いておりますが」
見る	同じ	「ただ、見てるだけでないのか！！ 窓辺にたち、海のほうをみました。

は 259 個である.

4.3　モデルによる評価実験

JWiC を用いて, 事前学習モデル BERT をファインチューニングし, 評価実験を行う. JWiC は, 2 文における同一語句が同じ語義で使用されているか否かを判定する 2 値分類タスクである. そのため, 入力文中のどの語句が対象であるかを明示する必要がある. そこで, BERT の埋め込みに対象語句を示す Target Embeddings を導入する. 図 7 に概要図を示す.

JWiC のデータは [train : validation : test] = [0.75 : 0.1 : 0.15] の比率で分割し, NICT BERT (BPE あり) をファインチューニングした. 表 13 において, JWiC の行が対象語句を明示しないモデル, JWiC$_{Target}$ の行が Target Embeddings を使用したモデルを表し, test セット列の JWiC が分類精度である. Target Embeddings を用いることでモデルが対象語句を認識し, 1.1%の精度向上が見られた.

4.4　JWiC の拡張

前節で構築した JWiC は 259 個の対象語句からなり, 読みづらさ検出に応用するには対象語句が少ない. そのため JWiC のデータ拡張に取り組む. データ拡張は, 追加する対象語句の選定, 例文の収集・ペアリング, 作成したペアのアノテーションという 3 段階で行う.

表 10: 使用したチェック設問の例

ラベル	語彙	例文 1	例文 2
違う	取る	電話を取るなり罵声。	要するに名より実を取る人種なのである。
違う	のばす	手をカップに伸ばす。	いまのは板ガネをローラーで延ばす だけでっしゃろ。
違う	できる	あなたも犬と話ができる！	むしろゆとりができたくらい」
違う	名	その名の由来がおもしろいのです。	要するに名より実を取る人種なのである。
同じ	理解	頭ではそう理解している。	しかし、この点が日本ではよく理解 されていない。
同じ	覚える	この頃は本で覚える。	それと、昔のお客さんは実際に食べて 料理を覚えた。

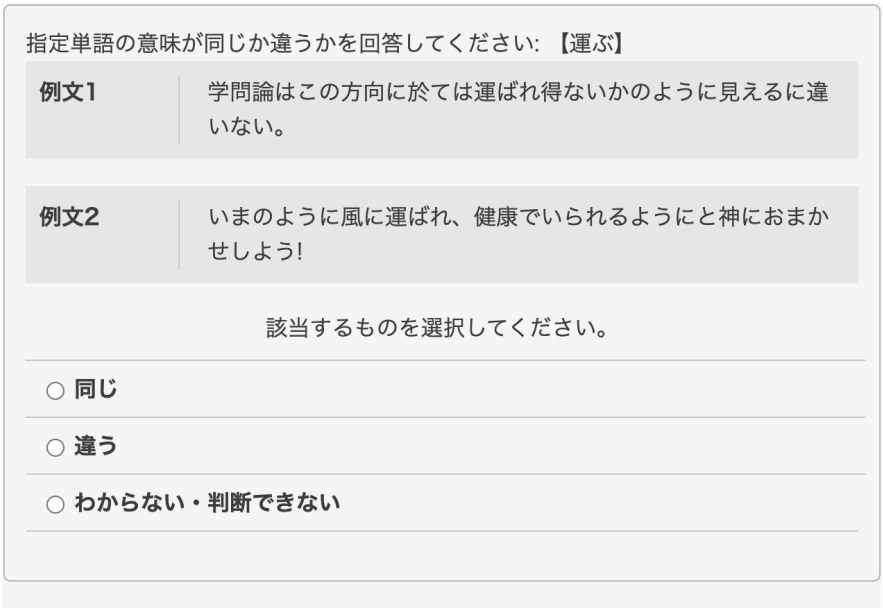


図 5: JWiC のクラウドソーシング

追加する対象語句の選定にあたり, 読みづらさ検出を執筆支援システムに応用することを考慮する. 筆記において使用される語句による拡大を行うために単語親密度を用いる. 対象単語の選別に単語親密度の「書く」の項目を使用し, ある程度筆記に使われる語句を採用し, かつ解釈に迷わないものは省くために 0.5 から 1.2 未満のものを対象とし収集した. 特殊文字を含むものやカタカナを含むもの, また, 2 単語以上からなるものを削除した. 対象をさらに使用頻度が高いものとするために, 品詞を動詞と形容詞に絞り, 最終的に 700 語句を収集した. このうち動詞が 442 語, 形容詞が 258 語である.

得られた対象語句に対して, 対象語句を 1 つのみを含む例文を「用例.jp[4]」から収集する. 収集した例文からペアを作成する. 同じ対象語句に対する例文ペア数は 10 ペアまでとし, JWiC にすでに含まれる対象語句に関しては数を減らすことで追加データに関してもバランス調整を行う.

得られた例文ペアに対して, 対象語句における語義が一致するか否かのアノテーションをクラウドソーシングで行う. 4.2 節と同様に, クラウドワーカーに [同じ, 違う, わからない・判

[4]https://yourei.jp/

表 11: クラウドソーシングの結果における正答率の平均と四分位数

文ペア	平均	最小	25%	50%	75%	最大
3,230	0.590	0	0.350	0.650	0.850	1.00

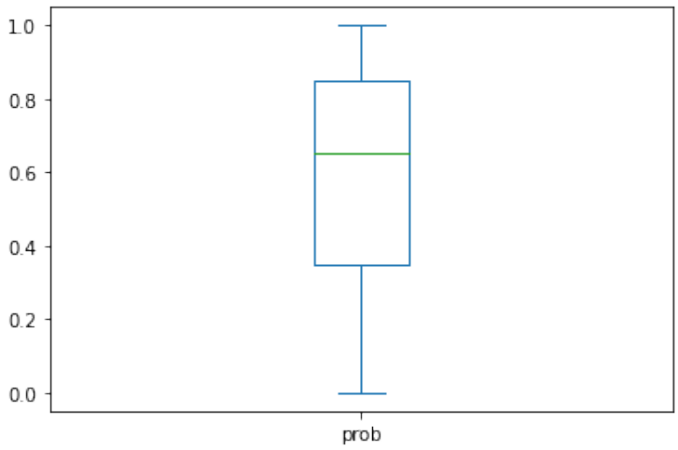


図 6: 正答率の箱ひげ図

断できない] の 3 択から選択してもらう. チェック設問は 4.2 節と同様のものを使用した. 作業画面は図 5 と同様である. 5,222 ペアに対して 10 人分の回答を収集した.

得られた回答集合を用いて, 回答比率を考慮した平均スコアに基づきラベル付けを行う. まず, [同じ, 違う, 不明] の各回答を [1, -1, 0] の数値と見做して, 10 人の回答の和をスコアとする. 閾値 m を使用し, $[-10, -m), [-m, m], (m, 10]$ の範囲において [違う, 棄却, 同じ] と分類を行う. m の値はラベルの偏りが大きすぎず, かつ判別のつかない文ペアが十分に除かれるように選択する. 本研究では閾値 $m = 2$ を採用した. 拡張したデータのラベル分布を表 12 に示す.

4.5　モデルの再学習と評価

データ拡張の前後における JWiC の精度変化を確認する. まず, 拡張したデータを学習データにのみ追加した場合の精度を表 13 の JWiC+ExTrain および JWiC+ExTrain$_{Target}$ の行に示す. データ拡張によって, 元の JWiC の対象語句についてはドメイン外の学習データが増えたが, Target Embeddings を用いたモデルは精度が落ちなかった.

次に, 拡張したデータを JWiC の train, validation, test セットに追加し評価した. この新しい test セットによる精度を表 13 の JWiC+Ex 列に示す. 新しい train セットで学習したモデルの精度を JWiC+Ex と JWiC+Ex$_{Target}$ の行に示す. 元の JWiC, JWiC$_{Target}$ モデルと比べて精度が向上していることが分かる. また, 各モデルにおいて Target Embeddings はやはり効果があることを確認した. データ分配の図解は図 8 に示す.

次節で述べる語義曖昧性判定による読みづらさ検出と評価においては, 拡張データを含めて学習した BERT モデル (Target Embeddings あり) を使用する.

図 7: Target Embeddings の導入

表 12: 拡張データのラベル分布

全ペア	同じ	違う	棄却	採用ペア
5,222 ペア	2,593	1,793	836	4,386 ペア

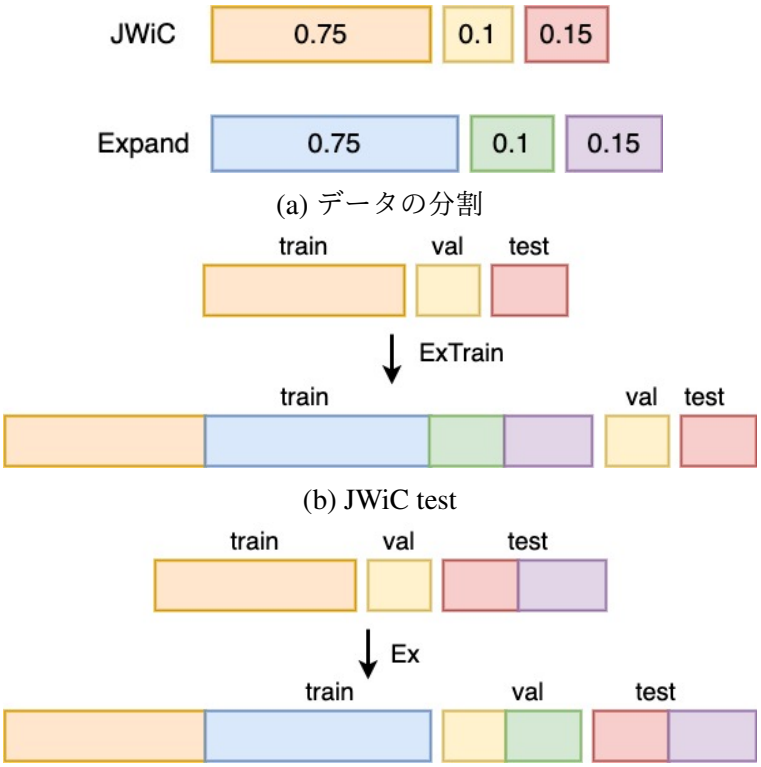
5　意味曖昧性に基づく読みづらさの検出

5.1　読みづらさの検出方法

読みづらさの検出は, 入力文中の対象語句が語義曖昧性を持つかを判定することによって行う. 語義曖昧性の判定は, 対象語句が持つ語義ごとに例文 1 つと入力文で文ペアを作り, WiC タスクを実施することによって行う. 複数の語義の例文と「同じ」と判定すれば複数の語義の可能性があり曖昧, 1 つのみと同じであれば曖昧でないとする. 図 9a の場合では, 「このいちごは甘い」という入力文は「とても甘いお菓子」のみと一致しており, 入力文中の「甘い」において使用されている語義が明白である. 一方, 図 9b の場合では, 傾向を示す「動き」と実際の活動を示す「動き」のどちらとも一致するため, どの語義で使用されているかが不明となり, 曖昧と判定される.

5.2　評価用データセットの作成と評価

JWiC から評価用データセットを作成する. 対象語句が持つ全ての語義との比較により評価するため, 各語義から例文を 1 文ずつ選択する. 2.3.2 節の例をとって考える. 「持つ」という単語は「Bringing」, 「Storing」, 「Possession」の 3 つの語義を持つ. この場合それぞれの語義を持つ例文を 1 つずつ用意する. そこに追加で「Bringing」の語義を持つ別の例文を用意し 4 つの文を 1 つのセットとする. このような形で語義例文と評価対象文のセットを作成する. これにより 184 セットを得た.



(a) データの分割

(b) JWiC test

(c) JWiC+Expand test

図 8: 学習データの分割と拡張

次に, 各セットにおいて評価対象文が語義曖昧性を持つか否かのアノテーションを行う. これは, JFN のフレーム情報を用いるのではなく, 一般的な人間による語義曖昧性の判断を得るために行う. 評価対象文に対して, どの例文が同じ語義であるかの複数選択式でクラウドソーシングにより 10 人分の回答を収集した. この回答をもとに 2 つの基準を設けて, 正解ラベルを作成した. 1 つ目は回答が分散していれば曖昧であり, 2 つ目は複数の語義の例文を選択していれば曖昧であるという基準である. まず, 最も回答数が多い選択結果 (セットで扱う) に対して n 人以上の回答が一致したものは回答が分散していないとみなし, n 人未満であれば分散しているため曖昧とみなす. n は 6 から 8 の値において試行した. 次に, n 人以上が一致した選択結果において, 複数の例文が選択された場合は曖昧であり, 単一であれば曖昧でないとする. フローチャートを図 10 に示す.

各閾値 n における曖昧か否かのラベル数と, 曖昧さ検出の精度を表 14 に示す. n=6, 7 においては 7 割程度の精度を達成しており, 語義曖昧性に基づく読みづらさの検出がある程度できた. n=8 において精度が低下したのは, 閾値が厳しいために, 分散しておらず, 本来曖昧でない回答についても曖昧であるというラベル付けとなったことが原因と考えられる.

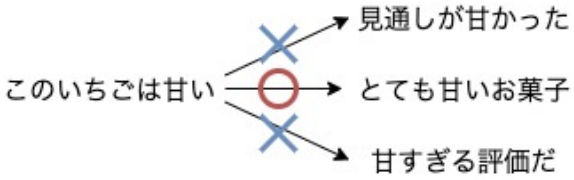
6 議論

6.1 難しい文と平易な文

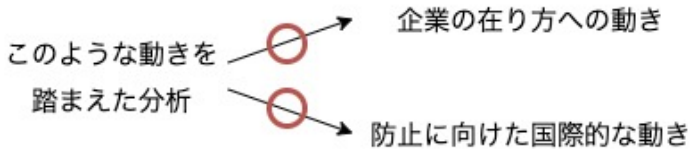
構造的に難しいと判別された文の特徴として主語が省略されているものや, 並列構造等が見られた. 文長が短いために一定数主語が無い文が存在しており, これらを排除することでより

表 13: 各モデルの分類精度

モデル	test セット JWiC	JWiC+Ex
JWiC	0.886	0.700
JWiC$_{Target}$	0.897	0.724
JWiC+ExTrain	0.872	-
JWiC+ExTrain$_{Target}$	0.900	-
JWiC+Ex	-	0.757
JWiC+Ex$_{Target}$	-	0.781



(a) 曖昧でない場合



(b) 曖昧な場合

図 9: 語義曖昧性検出の例

構造に焦点を当てた実験を行うことができると考えられる. また, 長い名詞句やカタカナは中身を読まずに塊であると判別できるため簡単と評価される傾向があった.

ここで, 図 11 にて難易度判定により有意な差を得た文における係り先未決定の動きを見る. 以下の文 (2a) は 10 人中 9 人が読みづらい文として選択し, 1 人が同等であると回答した.

(2) a. しかし、旧民社党は大半の議員が新進党に参加し、さきがけとの連携も流動的で連携相手は不確定だ。

b. 初期の警察署が置かれたセントラル地区のハリウッド通りには、インド料理店やインド系企業が多い。

(2b) は入力に適宜係り先が決定されているが, (2a) は保留されていることがわかり, 文の構造的難易度により未決定数に差が生じることが確認できる.

6.2　読みづらさの正解データ作成

多数決方式の n=5 においては [文 A, 同等, 文 B] の回答数が [1, 4, 5] の様な回答が存在し, 小さい n に対しては難易度の定まらない文が混在していると考えられる. 平均方式はこの欠点はカバーできるものの, 同等の文に対しての評価ができなくなる.

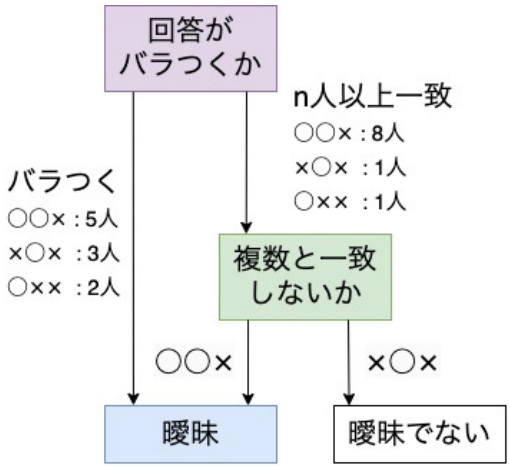


図 10: 語義評価のフローチャート

表 14: 閾値ごとのラベル分布と語義曖昧性の検出精度

閾値 n	6	7	8
曖昧	114	123	137
曖昧でない	70	61	47
精度	0.701	0.717	0.663

多数決方式にて n の値により収集される文ペアを見ていく. n=5 の例として (3), n=7 の例として (4) を示す. (3a) と (4b) が読みづらい文である.

(3) a. 九七年まで村山政権が続くのは安定かもしれないが、確信を持って政治がやれるのか。

b. この時は大分舞鶴がロスタイムに入って追いつき、抽選で長崎北陽台が決勝に進んだ。

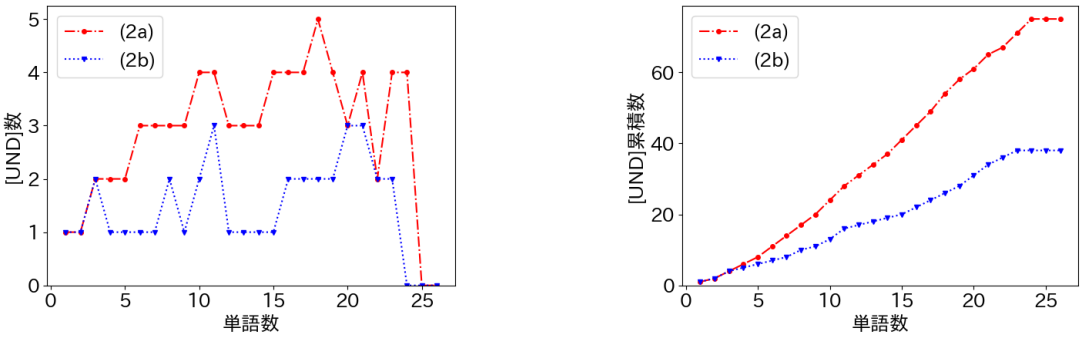
(4) a. 同県警では、県警山岳救助隊など約七十人で捜索したが、午後五時、捜索を打ち切った。

b. 本に囲まれ、埋もれ、重みで床が抜けそうだ、と心配するような生活にあこがれている。

(3) は文構造が似通っており難易度にさほど差が感じられないが, (4b) には並列構造が見られ, 係り先の理解に時間を要する. n を大きくすることで確かに難易度差がある文ペアを取得している.

6.3　校正における活用

構造的曖昧性の絶対評価は難しく, 本研究では 2 文の比較による評価を行った. 難易度のスコアはサプライザルと提案手法のどちらも各文ごとに付与している. そのため読みづらいと判別された文のスコアの平均・分散における解析が, その文単体での難易度の判別に繋がる. これを用いて難しい文の指摘を行うことで執筆者による修正や自動書き換え等の校正への活用が見込まれる.



(a) 各単語長でのカウント　　(b) 各単語長での累積

図 11: [UND] 数の比較

語義曖昧性においても複数の語義との比較という形で対象単語の読みづらさの判定を行った. 対象単語の語義数分の例文を用意すれば対象単語を拡張することができる. 加えて, 同義語辞書を用意することで代替単語の提案などへの応用に繋げることができる.

構造的曖昧性は語順, 語義曖昧性は多義語をそれぞれ対象とするため, 校正に活用するにあたってこれらは個別に扱うことを想定している.

6.4　モデルの改善

本研究では BERT による 2 値分類で JWiC タスクの分類を行った. Hugging Face Transformers[5] の BertForSequenceClassification を使用しており, 判定には [CLS] の Embedding を使用している. しかし, 提案した Target Embeddings の影響をより重視するためには, [CLS] 部分ではなく 2 文の TargetEmbeddings の足し合わせを予測に用いることでさらなる精度の向上につながると考えられ, 今後の課題とする.

7　終わりに

まず第一に, 日本語構造に着目した読みづらさの検出を行った. 漸進的構文解析による, 未入力文脈への係り構造を利用することで, サプライザルに比べて構造的に読みづらい文の検出を行うことができた. 今後は構文解析結果の top-K 比較による構文確率の比較等の拡張も検討する. また, 特定の曖昧性に的を絞った正解データの用意は困難であり, 収集方法も改善の余地がある.

第二に, JFN を使用した JWiC の構築及びデータ拡張を行い, これを応用して語義曖昧性に着目した読みづらさの検出を行った. 構造的曖昧性に続き語義曖昧性も絶対評価は難しく, 本研究では WiC タスクを語義の数だけ行う形で評価を行った. 収集した例文に対して語義の一致という 2 値分類のタスクに落とし込むことで, 語義ラベルを付与する必要性を排除した. これにより比較的容易にデータ拡張が可能となる. 評価においては語義ラベル自体は必要ないものの, 語義数分の例文収集が必要となる. 本研究では JWiC に含まれる対象語句を活用した評価を行ったが, 今後, 対象語句を拡張した評価を行うことが望まれる.

[5] https://huggingface.co/transformers

謝辞

日頃より熱心にご指導頂いた河原大輔教授に深く感謝する．また, 忌憚なき意見や助言を下さった研究室のメンバーにも感謝を表する.

日本語フレームネットを提供いただいた慶應義塾大学の小原京子教授に感謝する. 本研究は JSPS 科研費 JP21H04901 の助成を受けて実施した.

参考文献

[1] 村田匡輝, 大野誠寛, 松原茂樹. 読点の用法的分類に基づく日本語テキストへの自動読点挿入. 電子情報通信学会論文誌. D, 情報・システム = The IEICE transactions on information and systems, Vol. 95, No. 9, pp. 1783–1793, 09 2012.

[2] 宮地航太, 大野誠寛, 松原茂樹. 読みにくい語順の文への読点の自動挿入. 言語処理学会第 25 回年次大会 発表論文集, pp. 1308–1311, 2019.

[3] 田中麻祐子, 大野誠寛, 加藤芳秀, 松原茂樹, 石川佳治. 日本語推敲支援のための文の語順整序. 第 75 回全国大会講演論文集, Vol. 2013, No. 1, pp. 153–154, 03 2013.

[4] 髙須恵, 大野誠寛, 松原茂樹. RNNLM と SVM を用いた日本語文の語順整序. 第 82 回全国大会講演論文集, Vol. 2020, No. 1, pp. 453–454, 02 2020.

[5] Martin Schrimpf, Idan Blank, Greta Tuckute, Carina Kauf, Eghbal A. Hosseini, Nancy Kanwisher, Joshua Tenenbaum, and Evelina Fedorenko. The neural architecture of language: Integrative reverse-engineering converges on a model for predictive processing. bioRxiv, 2020.

[6] John Hale. A probabilistic earley parser as a psycholinguistic model. In Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies, NAACL '01, p. 1–8, USA, 2001. Association for Computational Linguistics.

[7] Roger Levy. Expectation-based syntactic comprehension. Cognition, Vol. 106, No. 3, pp. 1126–1177, 2008.

[8] 井上雅勝. 文の意味的曖昧性が構造的曖昧性の解消と保留に及ぼす影響 (2). 日本認知心理学会発表論文集, Vol. 2011, pp. 74–74, 2011.

[9] Mohammad Taher Pilehvar and Jose Camacho-Collados. Wic: the word-in-context dataset for evaluating context-sensitive meaning representations, 2018.

[10] 浅原正幸, 小野創, 宮本 エジソン正. BCCWJ-EyeTrack : 『現代日本語書き言葉均衡コーパス』に対する読み時間付与とその分析. 言語研究, No. 156, pp. 67–96, 2019.

[11] 浅原正幸, 加藤祥. 読み時間と統語・意味分類. 認知科学, Vol. 26, No. 2, pp. 219–230, 2019.

[12] 井上雅勝. 文の意味的曖昧性が構造的曖昧性の解消と保留に及ぼす影響. 日本認知心理学会発表論文集, Vol. 2010, No. 0, pp. 81–81, 2010.

[13] Masayuki Asahara. Between reading time and clause boundaries in japanese—wrap-up effect in a head-final language—日本語の読み時間と節境界情報—主辞後置言語における wrap-up effect の検証—. Journal of Natural Language Processing, Vol. 26, pp. 301–327, 06 2019.

[14] Masayuki Asahara. Between reading time and the information status of noun phrases 名詞句の情報の状態と読み時間について. Journal of Natural Language Processing, Vol. 25, pp. 527–554, 12 2018.

[15] 栗林樹生, 大関洋平, 伊藤拓海, 吉田遼, 浅原正幸, 乾健太郎. 日本語の読みやすさに対する情報量に基づいた統一的な解釈. 言語処理学会 第 27 回年次大会 発表論文集, pp. 723–728, 2021.

[16] 国立国語研究所. 分類語彙表 -増補改訂版-. 大日本図書刊, 2004.

[17] 浅原正幸. Bayesian Linear Mixed Model による 単語親密度推定と位相情報付与. 自然言語処理, Vol. 27, No. 1, pp. 133–150, 2020.

[18] 井上雅勝. ガーデンパス現象に基づく日本語文理解過程の実証的研究. 大阪大学博士論文, 2000.

[19] Tomohiro Ohno and Shigeki Matsubara. Dependency structure for incremental parsing of Japanese and its application. In Proceedings of the 13th International Conference on Parsing Technologies (IWPT 2013), pp. 91–97, Nara, Japan, November 2013. Assocation for Computational Linguistics.

[20] 大野誠寛, 松原茂樹. 漸進的係り受け解析の出力構造 -人間の文解析過程のアノテーション-. 言語処理学会 第 22 回年次大会 発表論文集, pp. 457–460, 2016.

[21] 後藤亮, 大野誠寛, 松原茂樹. 人間の漸進的言語処理能力の分析. 情報処理学会 第 82 回全国大会, pp. 457–458, 2020.

[22] Gerry Altmann and Mark Steedman. Interaction with context during human sentence processing. Cognition, Vol. 30, No. 3, pp. 191–238, 1988.

[23] George A. Miller. Wordnet: A lexical database for english. Commun. ACM, Vol. 38, No. 11, p. 39–41, nov 1995.

[24] Fuli Luo, Tianyu Liu, Zexue He, Qiaolin Xia, Zhifang Sui, and Baobao Chang. Leveraging gloss knowledge in neural word sense disambiguation by hierarchical co-attention. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 1402–1411, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[25] 谷田部梨恵, 佐々木稔. 訓練事例と辞書用例を異なるモデルで表現した語義曖昧性解消. 言語処理学会 第 27 回年次大会 発表論文集, pp. 957–960, 2021.

[26] 関谷洸, 佐々木稔. 語義の例文を使用した語義曖昧性解消の有効性分析. 言語処理学会 第28回年次大会 発表論文集, pp. 827–831, 2022.

[27] 曹鋭, 田中裕隆, 白静, 馬ブン, 新納浩幸. Bert を利用した教師あり学習による語義曖昧性解消. 言語資源活用ワークショップ発表論文集, pp. 273–279, 2019.

[28] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems, 2019.

[29] Alessandro Raganato, Tommaso Pasini, Jose Camacho-Collados, and Mohammad Taher Pilehvar. Xl-wic: A multilingual benchmark for evaluating semantic contextualization, 2020.

[30] Kyoko Ohara. Relating frames and constructions in Japanese FrameNet. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), pp. 2474–2477, Reykjavik, Iceland, May 2014. European Language Resources Association (ELRA). ACL Anthology Identifier: L14-1103.

[31] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[32] 柴田知秀, 河原大輔, 黒橋禎夫. BERT による日本語構文解析の精度向上. 言語処理学会 第25回年次大会 発表論文集, pp. 205–208, 2019.

[33] Xingxing Zhang, Jianpeng Cheng, and Mirella Lapata. Dependency parsing as head selection. In EACL 2017, pp. 665–676, 2017.