論文 / 著書情報
Article / Book Information

題目(和文)	高品質教師データが得られないドメインにおける要約手法の研究
Title(English)	
著者(和文)	狩野竜示
Author(English)	Ryuji Kano
出典(和文)	学位:博士（工学）, 学位授与機関:東京工業大学, 報告番号:甲第11758号, 授与年月日:2022年3月26日, 学位の種別:課程博士, 審査員:奥村 学,熊澤 逸夫,中山 実,篠﨑 隆宏,船越 孝太郎
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第11758号, Conferred date:2022/3/26, Degree Type:Course doctor, Examiner:,,,,
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

高品質教師データが得られないドメインにおける要約手法の研究

東京工業大学大学院

工学院情報通信コース

博士論文

指導教員：奥村 学 教授

狩野　竜示

2022 年 2 月

目次

図目次

表目次

第 1 章

序論

1.1 研究背景と既存研究の課題

インターネットおよびスマートフォンの普及により，日常的にデジタルテキストを読み書きする機会が増加している．総務省の公開する令和 3 年度版情報通信白書[*1]によると，日本国内におけるスマートフォンの世帯保有率は年々増加しており，2020 年には 85% を超えている．また，ソーシャルメディアやメッセージングサービスの利用率は 50% を超えており，若年層ほど利用率が高い．これらの事実は，人々がますます日常的にインターネット上のテキストに触れるようになっていることを示している．人々が触れる情報が増えるにつれ，素早く情報の取捨選択をするための技術の需要はますます高まってくると予想される．

その実現手段の一つが文書の自動要約技術である．要約は文書の内容を簡潔にまとめたものである．新聞記事の見出し，小説のあらすじ，科学論文の抄録など世の中には様々な形態の要約があり，情報を素早く取り入れ，行動の指針を決定することに役立っている．これを計算機によって自動化したものが自動要約である．

自動要約は古くから取り組まれてきた研究対象であり，1958 年には，重要単語を多く含む文を要約として抽出する手法が既に提案されている (Luhn 1958)．1969 年にはこれに加え，特定のキーワードや文の出現位置を考慮した手法が提案された (Edmundson 1969)．1980 年代には特定の知識を含むか否かのルールを組み合わせた方法で重要度をテキストに付与する手法が提案された (Fum et al. 1986)．ただし，これらは文同士の関係性を考慮せずに，文単体から重要度を判定する手法であった．2000 年初頭にはグラフベースの手法 (Mihalcea and Tarau 2004) や，特徴量ベクトルを使用した手法 (Radev et al. 2004) などが，文同士の関係性を考慮する教師なし手法として提案された．要約対象のテキストを本文と呼ぶが，自動要約には主

[*1] https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/r03/pdf/n1100000.pdf

に，本文から重要な文を抽出する抽出型要約と，言い換え表現などを用いて新たな文を再構成する生成型要約がある．前述したルールベースあるいは教師なし要約手法は全て抽出型要約手法であり，ニューラルネットワークの登場までは，生成型要約の研究はほとんど行われてこなかった．

Encoder と Decoder を利用する Sequence-to-sequence モデル (Sutskever et al. 2014) は提案されて以来，文から文を生成するタスクに幅広く用いられてきた．要約タスクも例外ではなく，Sequence-to-sequence モデルを利用した要約モデルは Rush et al. (2015) を嚆矢として様々なモデルが考案されてきた．Sequence-to-sequence モデルに使用される系列データモデルには Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber 1997) や Gated Recurrent Unit (GRU) が用いられてきたが，これらは並列計算ができないため，学習に時間がかかるという欠点があった． 並列計算が可能な系列データモデルである Transformer (Vaswani et al. 2017) の提案後はこれが Sequence-to-sequence モデルの主流となり, 要約タスクにおいても高い性能を発揮した (Ott et al. 2019)．その後，BART (Lewis et al. 2020) をはじめとした大規模事前学習言語モデルの登場により，生成型要約モデルの性能は更に向上した．

こうした要約手法の研究は，教師データが整備された一部のドメインに集中して研究が行われてきた．その代表例が新聞記事を対象としたものである (Nallapati et al. 2016)．これは，新聞記事の見出しを要約とみなすことで，本文から要約を生成，あるいは抽出する機械学習モデルを学習させることができるからである．教師なし学習は教師データを必要としないが，手法間の比較のため，教師なし要約の手法であっても新聞記事が対象とされることが多い (Zheng and Lapata 2019)．しかし，世の中で要約が望まれている文書は新聞記事に限らない．ソーシャルメディア，メール，チャット，医療文書など様々なドメインの文書が要約対象となりうる．こうした教師データが未整備のドメインの文書を要約する際の対処方法を 2 種類のアプローチに分けて論じる．

1 つ目のアプローチは，教師データを使用しない教師なし学習である．過去多くの教師なし要約手法が提案されてきた．既存の代表的な要約手法は，Centrality (Mihalcea and Tarau 2004; Erkan and Radev 2004; Zheng and Lapata 2019) や Centroid (Radev et al. 2004; Rossiello et al. 2017) を要約としてのふさわしさを表す指標として使用しているが，こうした手法は本文中で高頻度に言及される話題を要約として抽出しやすい．ここでは，言及頻度の高い話題は重要であるという仮定が前提にされている．ただし，実際には重要な話題が必ずしも多く言及されるとは限らない．そこで，新たな重要度の指標として，返信による言及のされやすさに着目する．メールやソーシャルメディアの投稿には，返信を送ることが可能である．こ

こで，返信によって言及されやすい話題は重要であるとの仮定に基づき，そうした話題を含む文を要約として抽出する手法を提案する．

2 つ目は低品質学習データから効率的に要約モデルを学習させる手法である．前述の通り，新聞記事は見出しを要約とみなし要約モデルの学習データとして使うことができる．新聞記事のようにメールやソーシャルメディアにおいても件名やタイトルを要約とみなすことができるため，一見同じ手法がメールやソーシャルメディアに対しても適用可能であるかのように思われる．ただし，新聞記事は文章執筆を生業とした記者が不特定多数の読者向けに執筆したものであるため，その質がある程度担保されている．それに対し，メールは個人間のプライベートなやり取りの中で書かれたものであり，ソーシャルメディアは匿名性の高い場で非業務的な活動として行われているため，質が低いものが多く含まれており，既存研究でも質の低さが指摘されている (Li et al. 2019; Zhang and Tetreault 2019)．要約は，本文の重要箇所を反映しているという前提があるが，メールやソーシャルメディアの件名，タイトルは，情報量が少なかったり，本文に無い情報を含むことがある．本論文ではこうしたノイズを多く含む要約データセットを用いて要約モデルを効率的に学習させる方法を提案する．

1.2 本研究の貢献

1.2.1 新たな指標に基づく教師なし手法の提案

本論文では，1 つ目のアプローチである教師なし学習手法の新たな手法を提案する．既存の教師なし抽出型要約手法では本文中の言及回数が多い話題を含む文を要約として抽出する．ただし，実際には重要な話題が必ずしも多く言及されるとは限らない．そこで新たな手法として，返信によって言及されやすい文を要約として抽出する手法を提案する．提案したモデルは，要約対象の本文と返信の対を用いて学習を行う．モデルは正しい返信とランダムサンプリングにより得られた偽返信を判別する学習を行う．その際，Gumbel Softmax により本文の一部の文を抽出し，正しい返信と偽返信の判別に使用する．モデルは学習の過程で返信によって言及されやすい文を本文から抽出するようになるため，評価時にはこれを要約として使用する．提案手法をメールデータセット及び，ソーシャルメディアデータセットで評価し，従来手法と同等もしくは上回る性能を発揮することを示す．

1.2.2 低品質要約教師データに対する効率的な学習手法の提案

本論文では，2 つ目のアプローチであるノイズを多く含む要約データセットに対する効率的な学習方法としてカリキュラムラーニングを要約データセットに利用した手法を提案する．カ

リキュラムラーニングは，ある指標に基づき学習するデータの順序を変更することでモデルの性能を上げる手法であり，ノイズを含むデータセットに対する有効性が示されてきた．今回，このカリキュラムラーニングを要約データセットに応用するにあたって，単一のノイズを含むデータセットからノイズを定量化する手法を提案し，既存手法よりも高い性能を発揮することをメールデータセット及びソーシャルメディアデータセットを用いて確認した．また，これまでカリキュラムラーニングに用いられてこなかった抽出率や含意判定確率が要約データセットに対するカリキュラムラーニングに有効であることも合わせて示す．

1.3 本論文の構成

本論文では 1.1 節 (研究背景と既存研究の課題) で述べた 2 つのアプローチに基づく新しい手法をそれぞれ提案する．第 2 章の関連研究では，上記 2 点の課題に対する対処としてこれまでにどのような関連研究が行われてきたかを論じる．第 3 章では，1 つ目のアプローチである教師なし学習に関して，既存の教師なし抽出型要約手法とは異なる要約の指標に基づいたモデルを提案する．第 4 章では，2 つ目のアプローチである低品質要約教師データに対する効率的な学習方法として，カリキュラムラーニングを要約データセットに利用した手法を提案する．第 5 章では第 3 章と第 4 章で提案した手法に対する結論および課題と今後の展望について述べる．

第 2 章

関連研究

第 2 章では本論文に関連する先行研究について議論する．まず，本論文の主要テーマである自動要約の先行研究について，教師あり手法と教師なし手法，評価方法，要約研究の対象となったデータセットに分けてそれぞれ議論する．要約手法には主に，本文から重要な文を抽出する抽出型要約と，言い換え表現などを用いて新たな文を再構成する生成型要約があるが, 教師あり要約手法に関しては, 第 4 章で取り扱う生成型要約手法について主に議論する．その後，第 4 章で議論する低品質学習データを使った要約モデルの学習に関連して，低品質学習データを使った学習方法一般の既存研究を議論する．

2.1　教師あり要約手法

要約分野は伝統的に教師なし手法が教師あり手法に比肩，もしくは上回る性能を発揮する稀有な分野であった．そのため，ニューラルネットワークベースの手法が開発される以前は，教師なし手法の開発が研究の主流であり，ニューラルネットワークベースの手法が考案されてからも，データセットによってはニューラルネットワークを使用しない教師なし要約手法の性能が，ニューラルネットワークベースの教師あり要約手法の性能を上回るケースが確認されている (See et al. 2017; Zheng and Lapata 2019)．これは，文の位置や他文書との類似性などの単純な要素が要約において重要であるということに由来すると考えられる．

ニューラルネットワーク以前の教師あり学習の手法は，文の重要度に関連する特徴量を設計したものが主流であった．文の長さや，“Conclusion”などの特定のキーワードの有無，段落の位置，大文字から始まる単語の有無などを特徴量にしたナイーブベイズ分類器 (Kupiec et al. 1995) や，テキスト中の位置や，固有表現の数，文の長さなどを特徴量に使用した Support Vector Machine モデルなどが提案されている (Hirao et al. 2002)．

教師あり手法が自動要約研究の主流になるのは Sequence-to-sequence モデル (Sutskever

et al. 2014) が提案されて以降である．Sequence-to-sequence モデルが提案される以前は，本文の内，重要な文を抽出する抽出型要約の研究が要約研究の主流であったが，Sequence-to-sequence モデルの登場により，本文に無い文章を新たに要約として生成する生成型要約の研究が活発化した．図 2.1 に Sequence-to-sequence モデルの模式図を示す．Sequence-to-sequence モデルは 2 つの系列データモデルを Encoder と Decoder として繋げたモデルであり，入力文を Encoder に入力すると，出力文が Decoder から出力される．入力文と出力文に該当するものはタスクによって異なり，翻訳タスクであれば，入力文は翻訳元のテキストであり，出力文は翻訳先のテキストとなる．要約タスクの場合，入力文は本文であり，出力文は要約となる．抽出型要約であれば，Decoder は各文を要約に含めるべきか否かの 2 値分類を行うことになるが，本節では，本論文第 4 章で扱う生成型 Sequence-to-sequence 要約モデルに限定して述べる．

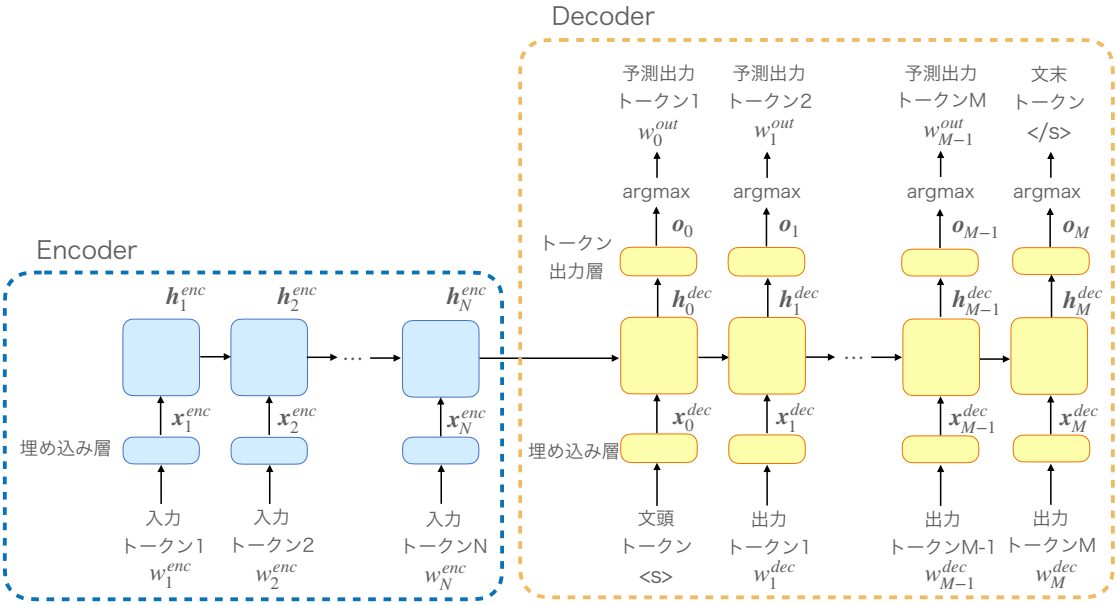

図 2.1　Sequence-to-sequence モデルの模式図.

Encoder は入力文の特徴量を計算する．まず，入力文は N 個のトークン (文字，または単語または Subword) に分割された後，辞書を通して 1 から V までの整数 $w_i^{enc}(i = 1...N)$ に変換される．辞書はトークンを 1 から V までの整数に対応付けるハッシュである．厳密には w_i^{enc} はトークンに対応する整数であるが，以後はこれをトークンと呼称する．トークンは埋め込み層を通じて，連続空間のベクトル $X^{enc} = \{\boldsymbol{x}_1^{enc}, \boldsymbol{x}_2^{enc}, ..., \boldsymbol{x}_N^{enc}\}$ に変換される．埋め込みベクトル X^{enc} を Encoder に入力すると，文脈を考慮した各トークンに対応する隠れ層 $\boldsymbol{h}_i^{enc}$ が得られる．

$$\boldsymbol{h}_i^{enc} = \text{Encoder}(X^{enc}) \tag{2.1}$$

次に，Decoder に Encoder の隠れ層 $\boldsymbol{h}_i^{enc}(i=1...N)$ と出力文のトークン $w_j^{dec}(j=1...M)$ を入力する．出力文は M 個のトークンから成るとすると，出力文のトークンは入力文と同様に埋め込み層を通じて連続空間のベクトル $X^{dec}=\{\boldsymbol{x}_1^{dec},\boldsymbol{x}_2^{dec},...,\boldsymbol{x}_M^{dec}\}$ に変換される．出力文のトークンを一つずつ入力していくと，Decoder は以下の式のように，入力トークンの次のトークンを予測する形で出力トークンの確率ベクトル $\boldsymbol{o}_j=\{o_{jk}\}_{k=1}^{V}$ を計算する．

$$\boldsymbol{h}_j^{dec}=\text{Decoder}(\boldsymbol{x}_j^{dec}|\boldsymbol{x}_1^{dec},...,\boldsymbol{x}_{j-1}^{dec},\boldsymbol{h}_1^{enc}...\boldsymbol{h}_N^{enc}) \tag{2.2}$$

$$\boldsymbol{o}_j=\text{softmax}(\text{W}\boldsymbol{h}_j^{dec}) \tag{2.3}$$

ここで，W は，$M\times V$ の行列であり，V は出力可能な単語数を指す．1 から V の整数は出力候補の各単語に対応している．最初に出力するトークンも予測しなければならないため，Decoder の最初の入力トークンは文頭を表す特殊トークン (通例：<s>) である．また，最後のトークンを Decoder に入力した時は，Decoder は文末を表すトークン (通例: < /s>) を出力するよう学習する．学習時には予測出力トークン w_j^{out} が次の出力トークン w_{j+1}^{dec} に一致するように最適化される．これは，w_{j+1}^{dec} の出力確率 $o_{jw_{j+1}^{dec}}$ が高くなるような Cross Entropy を損失関数に用いることで実現される．

$$L=-\sum_{j=0}^{M}\log(o_{jw_{j+1}^{dec}}) \tag{2.4}$$

学習時には Decoder には教師データとなる出力文を入力するが，推論時には，$\boldsymbol{o}_j$ の最も確率値が高いトークンを予測出力トークン w_j^{out} として出力し，次のトークンとして入力する．すなわち，推論時には $w_j^{dec}=w_{j-1}^{out}(j=1...M)$ となる．

Encoder 及び，Decoder には通例 Long Short-term Memory (LSTM) (Hochreiter and Schmidhuber 1997) や Gated Recurrent Unit (GRU) (Cho et al. 2014) が用いられていた．当初提案された Sequence-to-sequence モデル (Sutskever et al. 2014) では，Encoder で処理された入力文の情報は，固定長のベクトルとして，Decoder に入力された (通例，$\boldsymbol{h}_1^{enc}$ あるいは $\boldsymbol{h}_N^{enc}$，または両方を結合したベクトルが使われた．図 2.1 の模式図では $\boldsymbol{h}_N^{enc}$ を使用している．)．この場合，最初または最後のトークンに対応する隠れ層しか使われないため，入力文が長い時には情報が保存されず，Decoder にうまく伝達されないという問題があった．Attention (Bahdanau et al. 2015) はこの問題を解決する手法として提案された．系列データモデルは入力データの系列長と同じ数だけのベクトル $\boldsymbol{h}_i^{enc}$ $(i=1...N)$ を持つ．このベクトルは各系列における入力トークンと対応した情報を持っていると考えられる．Attention はこうしたベクトル群の内，Decoder の次の出力に有用なものを選択する機構であり，Decoder が使用できる入力データの情報が格段に広がり，様々なタスクで，従来の Sequence-to-sequence

モデルより高い性能を発揮した．Sequence-to-sequence モデルは学習に，膨大な数の教師データを必要とするが，見出しを要約とみなすことで，学習に成功させた研究 (Rush et al. 2015) が嚆矢となり，数多くの Sequence-to-sequence 要約モデルが提案されてきた．

生成型要約を含むテキスト生成モデル全般においては，未知語を出力できないという問題が存在した．図 2.1 に示したように，Sequence-to-sequence モデルは，トークン出力層において出力可能な単語の数 V と同じ次元のベクトルを出力し，softmax によって確率分布 $\boldsymbol{o}$ に変換，最も確率値が高い次元に該当する単語を出力する仕組みとなっている．ただし，計算量の関係上，あらゆる単語を出力することはできないため，トークン出力層の次元 V を数万程度に絞り，それ以外の単語は未知語トークン (通常は <UNK>) として出力することが一般的である．こうした処理を行う際，出力要約に含めたい単語が未知語である場合が往々にして発生し，精度低下の原因となっていた．コピー機構 (Gu et al. 2016; See et al. 2017) は，未知語の出力を可能にする機構として提案された．これは，入力文の各トークン位置に該当する隠れ層に Attention を貼り，その確率値に応じて入力文のトークンをコピーして出力する確率に反映する機構である．ただし，各単語を subword に分割することによって，未知語をなくす手法が提案されてからは，subword が未知語対処方法の主流になった (Sennrich et al. 2016).

既存の LSTM や GRU などの系列データモデルは，系列データ処理において，各系列のトークンを入力し，隠れ層の状態を更新した後でないと，次のトークンが入力できないという問題があった．これは並列処理が不可能なことを意味し，計算速度の低下につながった．これを解消するモデルとして，Transformer (Vaswani et al. 2017) が提案された．Transformer は多層ニューラルネットワークであり，各トークンに対応するベクトルと，他の全てのトークンベクトルとの Attention によって層を更新するモデルである．Transformer は Decoder においては，過去のトークンに対応するベクトルのみから Attention を計算するようマスキング処理を用いる．ここでは，Decoder の各トークンに対応するベクトル (図 2.1 における $h_j^{dec}(j = 1...M)$) は，同時に並列計算されるため，Transformer は学習時において，高速計算が可能である (推論時には，予測トークンを次の入力にするため，並列化はできない)．加えて精度面においても既存の LSTM や GRU の性能を上回っていたため，自然言語処理における新たな主流モデルとなった．要約タスクにおいても高い性能が確認されている (Ott et al. 2019).

2018 年 (国際会議の論文としては 2019 年) には大規模事前学習言語モデル BERT が登場し，言語処理のあらゆるタスクにおいて性能を大幅に向上させた (Devlin et al. 2019)．これは Masked Language Model と呼ばれ，入力文の一部をマスキングし，出力時にマスキングされた単語が何であったかを予測するタスクを解くことで事前学習を行う．この事前学習されたモ

デルを使って，別のタスクの Fine-tuning を行うと，高い性能を発揮する．BERT は双方向言語モデルであり，Sequence-to-sequence モデルのように，文を直接生成することができなかったため，Encoder を BERT とし，Decoder を通常の Transformer とした BERTSUM が生成型要約モデルとして提案された (Liu and Lapata 2019)．生成タスクに特化した Sequence-to-sequence モデルの大規模事前学習モデルとして，BART が提案され，要約タスクにおいても高い性能を発揮した (Lewis et al. 2020)．

本論文では，第 4 章で教師あり生成型要約モデルを用いた実験を行う．ここで使用するのは，先に紹介した 3 種類のモデルである．1 つは前述した古典的な Sequence-to-sequence モデルである Seq2seqWithAttention モデル (Bahdanau et al. 2015) である．これは Encoder と Decoder に LSTM を使用し，Attention 機構を採用したモデルである．2 つ目は Transformer (Vaswani et al. 2017), 3 つ目は事前学習済 Sequence-to-sequence モデルである BART (Lewis et al. 2020) である．

2.2 教師なし要約手法

要約手法は 2 つの方法に大別される．抽出型要約と生成型要約である．これまで提案されてきた多くの教師なし要約手法は抽出型である．教師あり要約においては，ニューラルネットワークを使用した手法が多く提案され発展を遂げているが，教師なし抽出型要約においては古典的手法が現在でも強力である．

教師なし要約手法で伝統的に使用されている強力な特徴量は，本文中で言及されている頻度である．最初の教師なし抽出型要約の手法は 1958 年に Luhn (1958) によって提案された．この時既に，前述した頻度が要約において重要な因子であることが述べられている．

> The “significance” factor of a sentence is derived from an analysis of its words. It is here proposed that the frequency of word occurrence in an article furnished a useful measurement for determining the significance of sentences.

後続の研究の多くも，本文中に言及されている頻度を要約に重要な指標としている．教師なし要約手法の多くの手法は，本文全体あるいは本文の各文との類似度が高い文を要約として抽出するが，これらの手法では，言及頻度の多い話題を含む文が抽出されやすい．類似度計算の方法には様々な手法があるが，代表的な手法に文間の類似度グラフの Centrality を用いた手法がある．Centrality は中心性とも訳され，グラフ上におけるノードの重要度を表す指標である．ここで言う重要とは，ハブになっている度合い，あるいは他のノードに与える影

響度の大きさなどを指し，Degree Centrality や Betweenness Centrality など様々な定義があるが，抽出型要約の文脈ではもっぱら Eigenvector Centrality の一種である PageRank (Brin and Page 1998) が用いられる．PageRank はある点 P がノード間をエッジの重みに応じた確率で一定時間 $(t \to t+1)$ 毎に移動する時，収束時に各ノードにいる確率を指す．点 P が時刻 t $(t \in \mathbb{N})$ にノード i にいる確率 $S^t(V_i)$ は以下となる．

$$S^t(V_i) = (1-d) + d \times \sum_{V_j \in In(V_i)} \frac{W_{ji}}{\sum_{V_k \in Out(V_j)} W_{jk}} S^{t-1}(V_j) \tag{2.5}$$

$In(V_i)$ はノード i にエッジを持つノードの集合を指し，$Out(V_j)$ はノード j がエッジを持つノードの集合を指す．d は dumping factor と呼ばれ，孤立したグラフがある場合に，点 P が永久にたどり着けないノードが出てしまうことを防ぐために，一定確率でノード i からノード j へ移動することを保証するための項である．この計算を繰り返し，S が収束した時の値が PageRank である．

TextRank (Mihalcea and Tarau 2004) はこれを抽出型要約に応用した手法 である．TextRank の文間類似度は以下の式で計算される．

$$W_{ij} = \text{similarity}(S_i, S_j) = \frac{|\{w_k | w_k \in S_i \cap w_k \in S_j\}|}{log(|S_i|) + log(|S_j|)} \tag{2.6}$$

ここで S_i, S_j は文，w_k は単語，$|S_i|, |S_j|$ は文 S_i, S_j にそれぞれ含まれる単語の数を表す．文間類似度に tf-idf を積算した Bag-of-Words ベクトルの cosine 類似度を用いた手法として LexRank (Erkan and Radev 2004) がある．元来の PageRank は Web ページ間を移動する確率をエッジとしたグラフに用いられていた．Web ページ A から Web ページ B に行く確率と，Web ページ B から Web ページ A に行く確率は異なるため，これは有向グラフであるが，通常文間の類似度に方向性は無いため，TextRank や LexRank で用いられる文間類似度グラフは無向グラフとなる．文同士の位置関係により，文 A から文 B へのエッジの重みと，文 B から文 A のエッジの重みを変えることで，文間類似度グラフに有向性を取り入れることで精度を大幅に向上させた PacSum と呼ばれる手法が提案されている (Zheng and Lapata 2019)．また，PacSum は，文間の類似度に分散表現を用いることで更に性能が向上することが示されている．

他に，教師なし抽出型要約手法における重要な概念として Centroid がある．複数ベクトルの Centroid は各ベクトルの平均であるが，正規化処理を行う手法では各ベクトルの和として表現される．MEAD (Radev et al. 2004) は，文をクラスタ毎に分割し，各クラスタの Centroid を計算し，各クラスタの Centroid に近い単語を多く含む文を要約として抽出する手

法である．この研究では，文間や単語間の類似度を tf-idf を用いて計算しているが，後続の研究 (Rossiello et al. 2017) では，これを Word Embedding (Mikolov et al. 2013) に置換した手法が提案されている．ベクトルの平均が要約として表現されるという発想は，後段の研究にも受け継がれ，複数文の特徴量ベクトルの平均ベクトルから要約を生成する MeanSum (Chu and Liu 2019) や，Variational Auto-Encoder (VAE) の事前分布の平均ベクトルから要約を生成する CopyCat (Bražinskas et al. 2020b) が提案されている．

その他にも，抽出文の特徴量ベクトル（単語の頻度ベクトルや Paragraph Vector などを使用）から本文の特徴量ベクトルを復元する Reconstruction Loss (He et al. 2012; Liu et al. 2015; Ma et al. 2016) を用いたものや，抽出文と文全体の Unigram 単語分布の Kullback-Leibler divergence (Haghighi and Vanderwende 2009) を利用した手法がある．単語をノードとしたグラフの経路スコア計算を用いた手法 (Mehdad et al. 2014; Shang et al. 2018) は，複数文圧縮アルゴリズム (Filippova 2010) に依拠している．

上述した従来の教師なし抽出型要約手法は，前述した通り，重要なトピックは高頻度に言及されるという前提に基づいて設計されている．図 2.2 にあるように Centrality ベースの手法や Centroid ベースの手法は他の文との類似度が平均的に高い文を要約として抽出する．こうした手法は，高頻度に言及されるトピックを含む文を要約として抽出しやすい．但し，実際には言及回数が低いトピックが重要であることはありうる．本論文第 3 章では，言及頻度に依存しない新しい重要度の指標を提示する．

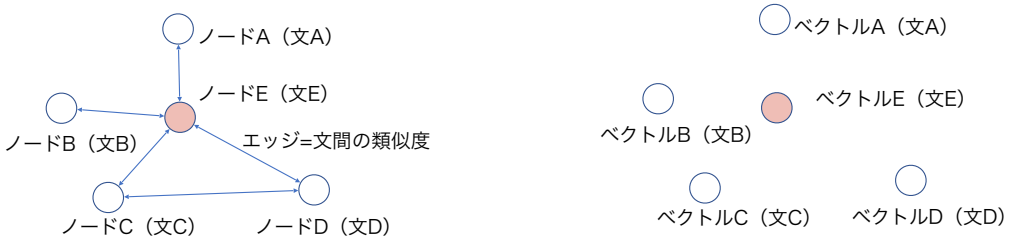

図 2.2　(左) 文をノード，文間の類似度をエッジとしたグラフ．Centrality ベースの手法では，他の文との類似度が高い文が要約として抽出される（赤ノード）．(右) 文のベクトル表現．Centroid ベースの手法では，全体の平均に近い位置にある文が要約として抽出される（赤ノード）．

深層学習の登場以降は，教師なし学習の用法が変化した．深層学習の利点として，人手による設計なしで特徴量が自動的に設計されることが挙げられている (LeCun et al. 2015) が，これは逆の見方をすると教師がなければ特徴量が作られないことを意味している．自然言語処理の分野においては，単語を数十から数百次元のベクトルに置き換える Embedding 手法がニューラルネットワークベースの手法として用いられている．こうしたベクトルは学習後に単

語の特性を表すようになるが，学習が行われていない段階では単なる乱数の羅列になってしまう．単語の特徴量を学習させるために，容易に教師データが得られるタスクとして，前後の単語の予測や，マスクされた単語を予測するタスクが用いられる．こうした手法は厳密には教師あり学習ではあるが，ニューラルネットワークには上述したように，教師なしでは特徴量が取得できないという特性があるため，教師なし学習と呼ばれている．すなわち，ニューラルネットワークの台頭以降の教師なし学習は，事前学習により得られた特徴量を伝統的な教師なし学習手法に適用したものと，容易に得られるデータを教師とした End-to-End の学習手法の二種類に大別できると考えられる．

前者の，事前学習で得られた特徴量を利用した教師なし学習手法は，抽出型要約手法がいくつか提案されている．Skipgram から得られた単語ベクトルの和を文ベクトルとして，サブモジュラ最適化を行う手法 (Kågebäck et al. 2014) や，Paragraph Vector を用いて計算された抽出文と全文の特徴量の誤差を最小化する手法 (Ma et al. 2016)，CNN 言語モデルや BERT の特徴量を TextRank に適用する手法 (Yin and Pei 2015; Zheng and Lapata 2019) が提案されている．これらは事前学習されたニューラルネットワークを特徴量抽出器として用いて，既存の要約モデルに適用しているが，本論文第 3 章では End-to-end での学習が可能な手法を提示する．

後者の End-to-end で学習可能なニューラルネットワークベースの要約モデルは，生成型のモデルがいくつか提案されている．文圧縮においては，シャッフルされた単語を元の順番に並び替えるタスクを使用したもの (Fevry and Phang 2018)，圧縮された文から元の文を復元するタスクを使用したもの (Baziotis et al. 2019) がある．レビュー文の生成型要約として，木構造の親ノードから要約生成を行うもの (Isonuma et al. 2019)，レビュー文の特徴量ベクトルの平均ベクトルから要約を生成する手法 (Chu and Liu 2019)，VAE（Variational Auto-Encoder）の事前分布を要約生成に応用したもの (Bražinskas et al. 2020b) がある．他に，マスクされた文を復元するタスクを要約生成に応用した研究がある (Laban et al. 2020)．本論文の第 3 章で提案する手法はこちらの End-to-End 型に属する．

メール，チャット，ソーシャルメディアなどのオンライン会話の要約は古くから研究されている．ニューラルネットワークを利用した要約モデルが台頭している一方，会話文の要約には非ニューラルモデルが使われている．既存手法として，単語をノードとしたグラフの経路スコア計算を利用したもの (Mehdad et al. 2014; Shang et al. 2018) がある．Dialogue Act 分類は，文をその役割（例：質問，回答，挨拶など）ごとに分類するタスクであるが，会話文の要約研究にも応用されている (Bhatia et al. 2014; Oya and Carenini 2014)．

引用は要約の重要な要素になりうる．我々は返信する際，重要な箇所を強調するため，引用

を利用する．引用を特徴量として活用した研究として，引用に出現した単語に重み付けをして，Centroid ベースの手法を改善したものがある (Carenini et al. 2007; Oya and Carenini 2014)．この先行研究では，引用を補助的な特徴量として使用したが，本論文では，第 3 章で引用のみに着目した手法を提案する．また，提案するモデルは明示的な引用を教師信号として使わずに，暗黙的引用を抽出する．

2.3 要約モデルの評価方法

要約モデルの自動評価指標にはもっぱら ROUGE (Lin 2004) が使用されてきた．これはモデルが出力した要約と参照要約の単語の一致率によって計算する指標である．n-gram の一致度を計算する ROUGE-n や，最長共通部分列を計算する ROUGE-L がある．ROUGE は Recall-Oriented Understudy for Gisting Evaluation の略であり，その名の通り，従来は Recall ベースの指標として提案されたが，近年では長い文が有利になってしまうことを防ぐために F1-score で評価を行うのが一般的である．ROUGE-n の Precision (ROUGE-n-P), Recall (ROUGE-n-R)，F1-score (ROUGE-n-F) はそれぞれ次式で定義される．

$$\text{ROUGE-n-P} = \frac{Count(\text{出力要約中の n-gram} \cap \text{参照要約中の n-gram})}{Count(\text{出力要約中の n-gram})} \tag{2.7}$$

$$\text{ROUGE-n-R} = \frac{Count(\text{出力要約中の n-gram} \cap \text{参照要約中の n-gram})}{Count(\text{参照要約中の n-gram})} \tag{2.8}$$

$$\text{ROUGE-n-F} = \frac{2 \times \text{ROUGE-n-P} \times \text{ROUGE-n-R}}{\text{ROUGE-n-P} + \text{ROUGE-n-R}} \tag{2.9}$$

ROUGE-L においては, 最長共通部分列の長さを出力要約の長さで割ったものが ROUGE-L − P となり, 参照要約の長さで割ったものが ROUGE-L-R となる. 生成型，抽出型ともに，ROUGE-1, ROUGE-2, ROUGE-L を用いるのが一般的であり，本論文でもこれらを用いて評価を行う．

ROUGE は特に生成型要約において，人手評価との相関が低いことが指摘されている (Kryscinski et al. 2019)．そのため，生成型要約モデルを用いる本論文第 4 章では ROUGE での自動評価に加えて人手評価を実施する．人手評価においては，重要な情報が表現されているか，文章が流暢であるか，本文と矛盾する記述がないかなどの観点で評価される．本論文では，この内，情報量と流暢性を人手評価に使用する．

2.4 要約研究の対象となったデータセット

本節では，自動要約の研究対象としてどのようなドメインのデータセットが用いられてきたかを概観する．伝統的にニュース記事はよく要約データセットの対象とされてきた．2000 年代前半には文書要約のデータセットが国際会議 Document Understanding Conference (DUC) [*1]で公開された．DUC は 2001 年から 2007 年まで行われたが，一環してニュース記事の要約データセットが公開されている（2001 年と 2002 年には論文の要約データセットも公開されている）．2008 年から 2011 年，2014 年には Text Analysis Conference (TAC) [*2]で要約データセットが公開タスクとして取り組まれてきたが，そこでも対象とされたのはニュース記事であった (2008 年，2010 年，2011 年)．ただ，2008 年にはブログ記事，2014 年には医療文書が対象とされている．TAC は 2015 年以降も開催されているが，2015 年から 2020 年にかけては要約タスクは公開タスクに含まれなくなった．

上記のデータセットは，小規模な 100 から数 100 程度のテキストに対し，人手で要約を作成したデータセットである．これらは多くの場合評価用のデータセットとして使われた．また人手で作成した要約は複数人のアノテーターによって作成されることが多かった．深層学習以降は，Web ページをクロールしたデータを学習データ及び評価データとして使用することが多くなった．それに伴い，要約データセットは数万から数十万のデータ量へと大規模化し，また正解とされる参照要約は 1 つのみを用いることが多くなった．

要約タスクにニューラルネットワークを利用した研究の嚆矢である Rush et al. (2015) の研究では，大規模ニュース記事コーパス Gigaword の見出しを要約とみなして要約モデルの学習データとした．これ以降，新聞記事のデータが要約の学習データとして使われることが一般的になり，CNN Daily Mail (CNNDM) (Hermann et al. 2015) や New York Times (NYT) (Sandhaus 2008) が要約研究に多く用いられてきた．近年では新聞記事の他にも，多様なドメインのデータセットが要約タスクの対象となっており，論文 (Cohan et al. 2018) や，特許公報 (Sharma et al. 2019)，Wikipedia (Koupaee and Wang 2018)，ソーシャルメディア (Kim et al. 2019)，レビューサイト (Li et al. 2019)，メールデータセット (Zhang and Tetreault 2019)，対話テキスト (Zhao et al. 2020)，国会法案 (Kornilova and Eidelman 2019) などが要約データセットとして構築，使用されている．

[*1] https://duc.nist.gov/
[*2] https://tac.nist.gov/

2.5 低品質教師データに対する効率的な学習方法

■学習データの低品質性 機械学習の学習データとして使われているデータセットは低品質なデータを多く含むという指摘は多くされてきた．深層学習はモデルの構造上多量の学習データを必要とするため，深層学習の研究が主流になるにつれ，Web から自動で大量に収集したものが学習データとして使われることが多くなった．代表的な例が画像分類タスクで用いられている ImageNet (Deng et al. 2009) や Webvision (Li et al. 2017) である．これらは Web で検索して得られた画像を人手でアノテーションしたデータセットであるが，数多くのラベル誤りを含むことが報告されている (Li et al. 2017; Northcutt et al. 2021)．こうしたラベル誤りなどの本来の教師信号にそぐわない低品質なデータを先行研究ではノイズと呼称しているため，本論文でも同様にノイズという表現を用いる．

自然言語処理の分野においても，Web から収集した大量のデータをモデルの学習に使う試みが成されている．関係性抽出において，Freebase に記載されている関係性データと同じ情報を含むテキストを Wikipedia から抽出し，それを学習データとする研究がある (Mintz et al. 2009)．後続の研究によって，これらは多くのノイズを含むことが指摘されている (Qin et al. 2018)．翻訳タスクにおいては，Web から収集された多言語（主にヨーロッパの言語）パラレルコーパス ParaCrawl (Esplà et al. 2019) が存在するが，これにはノイズが多く含まれることが指摘されている (Wang et al. 2018)．ここで言うノイズは，アラインメントの失敗により翻訳元文あるいは翻訳先文に互いに対応していない情報が含まれていることを指す．

Sequence-to-sequence 要約モデルの学習においては，通常，見出し，タイトル，件名などを要約とみなして学習する．しかしながら，これらのデータの内，メールやソーシャルメディアの件名およびタイトルは，非公式な場や，匿名性の高い場で書かれたものであるため，品質が担保されていない． Zhang et al. (2019) はメールの件名を生成するタスクを提唱したが，元の Enron コーパスに含まれている件名がノイズを多く含んでいたため，新たにノイズの少ない評価データセットを構築している． Li et al. (2019) は，ルールや分類モデルを使ってレビューデータのノイズをフィルタリングした．新聞記事の見出しは，新聞社が不特定多数の読者に公開するものであるため，比較的要約として適切なものが多いものの，本文から推測が不可能な情報が含まれることが指摘されており，含意判定モデルによってそうしたデータをフィルターする手法が提案されている (Matsumaru et al. 2020)．

■低品質データでの学習方法 低品質データに対する学習方法は過去多く研究されており，多くは画像分類タスクを対象に行われている．モデルはノイズを含むデータに対しては，一定の

予測結果を出さないという知見に基づき，複数のモデルのアンサンブルを利用して，ノイズデータを検出する手法の有効性が示されている (Lee and Chung 2020; Nguyen et al. 2020; Tarvainen and Valpola 2017)．モデルのパラメータに摂動を加えた複数のモデルのアンサンブルを使用する手法 (Lee and Chung 2020) や，学習中に保存された Epoch ごとのモデルのアンサンブルを使用する手法 (Tarvainen and Valpola 2017; Nguyen et al. 2020)，学習時の Random Seed を変えたモデルのアンサンブルを使用する手法などが存在する (Meng et al. 2021)．

損失関数を改良して，ノイズに対処した先行研究も存在する．一般的に用いられている平均二乗誤差 (MSE: Mean Square Error) や Cross Entropy と言った損失関数がノイズに脆弱であるという報告が成されており (Zhang and Sabuncu 2018; Feng et al. 2020; Ma et al. 2020; Wang et al. 2019)，これを改善するためノイズに頑健な損失関数を提案した研究がある (Lin et al. 2020; Zhang and Sabuncu 2018; Wang et al. 2019; Ma et al. 2020)．Wang et al. (2019) は，真のラベル分布関数と，予測分布関数を入れ替えた Reverse Cross Entropy (RCE) Loss を提案し，それがノイズに対して頑健であることを示した．更に，RCE Loss を通常の Cross Entropy Loss の組み合わせて学習させる Symmetric Cross Entropy Learning を提案し，ノイズを含むデータから学習したモデルの性能が上がることを示した．Ma et al. (2020) は，Cross Entropy や Mean Absolute Error などのあらゆる損失関数が正規化処理によってノイズ頑健性を得ることを示した．

アンサンブルや損失関数を用いた手法の他に，ノイズを吸収する層を追加する手法 (Goldberger and Ben-Reuven 2017) がある．Early Stopping は過学習を防ぐ手法として広く使われているが，ノイズを含むデータに対しても有効であることが示されている (Li et al. 2020)．

上記の手法は，トークン単位で生成を行うテキスト生成タスクにそのまま適用することは困難である．アンサンブル手法は，複数のモデルである程度一致する分類結果を出力することを前提としているが，テキスト生成においては出力文がモデル間で一致することは稀である．またテキストデータは，あるトークンを別のトークンに置換すれば，学習データを修正できると言った性質のものではないため，トークン単位でのラベル誤りを前提とした損失関数ベースの手法も適用できない．

テキスト生成タスクにおいては代わりに，ノイズの多い学習データをフィルタリングする手法や，カリキュラムラーニングを使う手法が提案されている．学習データのフィルタリングを行う方法として，対話タスクにおいて，entropy を用いて dull response (当たり障りの無い応答) を学習データから除去した手法が提案されている (Csáky et al. 2019)．翻訳タスクにおいては，多言語事前学習言語モデルを用いて，対応関係の低いデータを除去する手法が提案され

た (Zhang et al. 2020)．こうしたフィルタリングを行う手法には閾値設定の困難性の問題が存在する．フィルタリングは，ある指標の閾値を上回るあるいは下回るかを基準に行われるが，どの閾値が最適であるかを明らかにするには閾値の設定の数だけ学習を繰り返す必要がある．こうした問題があるため，本論文ではカリキュラムラーニングに着目する．

■カリキュラムラーニング　カリキュラムラーニングは学習データの順序を変更することで収束速度やモデル性能を上げる手法である (Bengio et al. 2009a)．過去の研究 (Cirik et al. 2016) はこれを文生成タスクに応用し，2 種類のカリキュラムを提唱した．Baby step カリキュラムと One-Pass カリキュラムである．後続の研究がカリキュラムラーニングを翻訳タスクに応用したが (Kocmi and Bojar 2017; Platanios et al. 2019; Wang et al. 2019; Zhou et al. 2020)，要約タスクに応用した研究はいまだ存在しない．

カリキュラムラーニングは元来，難易度で学習データをソートする手法であった．しかし，近年ではノイズの多さでソートを行う手法が提唱されている． Wang et al. (2018) は，2 つのモデルを使って学習データのノイズを定量化する手法を提案した．同様のアルゴリズムを使い， Kumar et al. (2019) は強化学習を用いて学習に適している学習データのセグメントを適宜選択していく手法を提案した．Wang et al. (2019) はノイズ定量化に加えて，ドメインらしさを定量化し，EM アルゴリズムを用いてそれらを組み合わせるカリキュラムラーニングの手法を提案した．要約分野においては，同じドメインでノイズの多寡が異なるコーパスは存在しないため，前述のノイズ定量化手法を要約モデルに適用することはできない．そのため，本稿ではノイズを含む単一コーパスからノイズを定量化する手法を提案する．

第 3 章

会話文における暗黙的引用を利用した抽出型教師なし要約

3.1 概要

インターネット上の会話が活発化するにつれ，会話文の自動要約技術の必要性は益々増している．ニューラルネットワークを使用したモデルは教師あり要約において，高い性能を発揮しているが，教師なし要約への応用は未だ限定的である．教師あり要約モデルの学習には，数万の要約-本文対が必要になる．あらゆるドメインにおいてこれらの対データを用意することは現実的ではないため，教師なし要約の手法が求められている．我々は返信を伴う会話形式のテキストを対象とした教師なし要約手法を提案する．

過去，多くの教師なし要約手法が提案されてきた．文の類似度グラフの Centrality を使用した手法は強力な教師なし要約手法であり (Mihalcea and Tarau 2004; Erkan and Radev 2004; Zheng and Lapata 2019)，会話文の要約にも応用されている (Mehdad et al. 2014; Shang et al. 2018)．Centrality の他にも，文の特徴量ベクトルの Centroid (Gholipour Ghalandari 2017)，Kullback-Leibler divergence (Haghighi and Vanderwende 2009)，Reconstruction Loss (He et al. 2012; Liu et al. 2015; Ma et al. 2016)，単語をノードとした有向グラフの経路スコア計算 (Mehdad et al. 2014; Shang et al. 2018) などが，要約に使われている．上記全ての手法の前提にあるのは，重要なトピックは文書中に高頻度に言及されるという点である．しかし，重要なトピックは必ずしも高頻度に言及されるわけではない．そのため，もし重要なトピックの言及回数が少ない場合，上記手法は重要文の抽出に失敗する．より高精度の要約を実現するためには，“頻度”とは異なる文書の側面に着目する必要がある．

頻度とは異なる文書の重要度の指標として，我々は “引用のされやすさ”に着目する．我々

は，メールや投稿文に返信する際，投稿の一部分を引用することがある．具体例を図 3.1 に示す．右側の返信の例にあるように．引用文は，引用符 “ >”から始まり，返信先の投稿文中の文・フレーズと一致する箇所を指す．高頻度に引用される箇所は重要であると考えられるため，引用される箇所を予測できれば，本文中で言及される頻度に関わらず，重要な情報を含む文を抽出できると考えられる．過去の研究に，引用を要約モデルに補助的に利用したものがある．Carenini は，引用文に現れる単語に重み付けをし，Centroid ベースの要約手法の精度を向上させた (Carenini et al. 2007; Oya and Carenini 2014)．ただし，ほとんどの返信は明示的な引用を含まない．そのため，引用を直接教師データとして扱うことは難しい．我々は，引用文を教師として使用せずに引用箇所を抽出できるモデル，Implicit Quote Extractor（IQE：暗黙的引用抽出器）を提案する．図 3.1 に示す例のように，引用文は返信が言及している投稿の箇所であるため，明示的な引用が無い場合にも，返信内容から本来引用されるべき箇所を間接的に特定できる．これを暗黙的引用と呼称する．

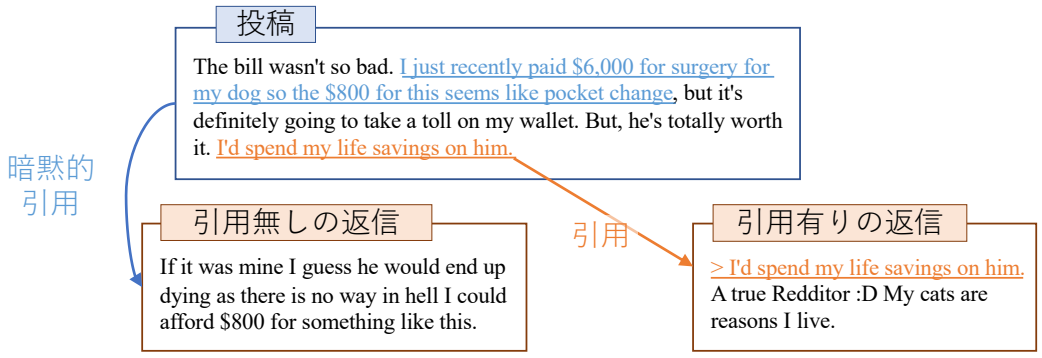

図 3.1　投稿と引用付きの返信と引用無しの返信の例．暗黙的引用は，返信が言及している投稿の一部であるが，返信には明示的に示されていないものを指す．

IQE は，返信によって言及される箇所を特定することにより，明示的な引用なしに，引用箇所を抽出することを目指す．IQE は投稿と返信候補のみを学習に使用し，明示的な引用を使用しない．返信候補は，投稿に対する実際の返信，あるいはランダムにサンプリングされた返信である．学習タスクは，返信候補が実際の返信であるかどうかを判定することである．IQE は，投稿から少数の文を抽出し，これを真偽判定の特徴量に使う．IQE は返信候補の真偽判定の性能を向上させるように文抽出のパラメータを学習するので，返信が言及しやすい文を要約として抽出するようになる．要約は本文のみから作成する必要があり，返信に依存してはならない．そのため，IQE は抽出文の選択に，返信の特徴量を使わない．すなわち，IQE は学習時にのみ返信を必要とし，評価時には必要としない．IQE が抽出するのは返信に依存した引用箇所では無く，返信によって最も引用されやすい箇所となる．

IQE を 2 つのメールデータセット，Enron 要約データセット (Loza et al. 2014) の業務メールと私用メールで評価し，また，ソーシャルメディアのデータセットとして，Reddit TIFU データセット (Kim et al. 2019) でも評価を行い，多くのベースラインの性能を上回ることを確認した．

提案したモデルは 2 つの仮説に基づいている．1 つは提案モデルが引用を抽出できるという点である．IQE は引用抽出を目的としているが，引用を教師として使用していないため，実際に引用される文を抽出できるかは明らかでない．そのため，我々は，提案モデルがどの程度引用を抽出できるか評価する．もう 1 つの仮説は，引用は要約として有用であるという点である．先行研究 (Carenini et al. 2007; Oya and Carenini 2014) は引用文を利用して，Centroid 要約モデルの性能を向上させ，引用が要約に有効であることを示した．しかしながら，これらの先行研究は引用を補助的な特徴量として使用しているため，引用それ自体が要約になりうるかは明らかでない．これを検証するため，我々は引用を要約とみなし，その ROUGE 値を評価することで，引用が要約として有用であるかを評価する．

引用が多数存在する Reddit データセットで，上記 2 点の仮説を検証し，仮説を裏付ける結果を得た．また，定量的，定性的 2 つの観点で，頻度ベースの既存手法が抽出できない重要文を提案モデルが抽出できることを示した．

本章の研究の貢献は以下の 3 つである．

- 言及頻度に依存した従来の教師なし抽出型要約手法の問題点を指摘し，新たな文書の重要度の指標として“返信による引用のされやすさ”を提案，実験により有効性を示した．
- End-to-end で学習可能な教師なし抽出型要約モデル，Implicit Quote Extractor（IQE）を提案し，ベースラインと同等の性能を示すことを 2 つのメールデータセットと 1 つのソーシャルメディアデータセットを対象にした評価実験によって示した．
- 引用を実際に含むソーシャルメディアデータセットを使い，提案モデルが引用を抽出しやすいこと，また，引用が要約に有用であることを示した．

3.2 提案モデル

本章の研究では，教師なし抽出型要約モデルである Implicit Quote Extractor (IQE) を提案する．図 3.2 にモデルの概要を示す．学習時のモデルの入力は投稿と返信候補である．返信候補は，投稿に対する真，あるいは偽の返信である．IQE の学習タスクは返信候補が真であるか偽であるかを判定することである．

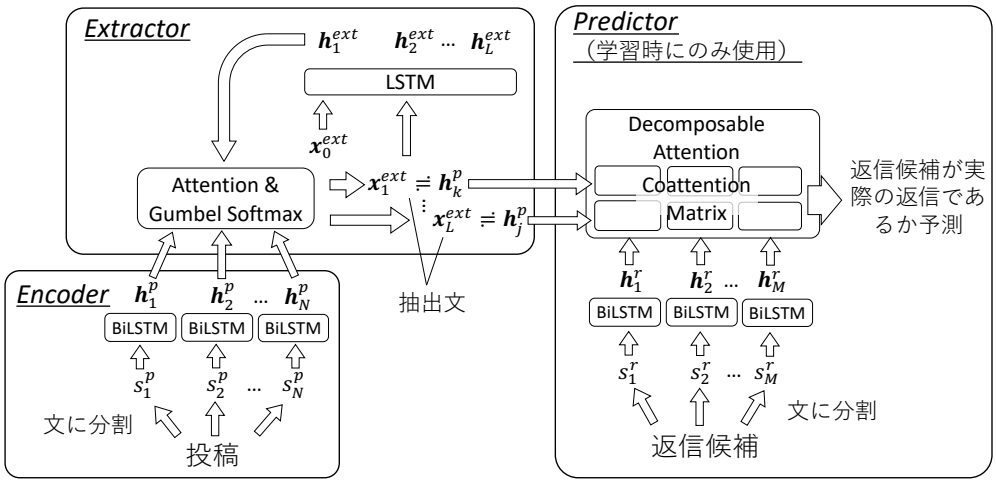

図 3.2　提案モデル，Implicit Quote Extractor (IQE) の概要図．Extractor は，投稿の文をいくつか抽出し，返信候補が投稿に対する真の返信であるかの予測に利用する．k と j は 1 から N の整数である．

IQE は 3 つの構成要素から成る．Encoder，Extractor，Predictor である．Encoder は，投稿文の特徴量を計算する．Extractor は少数の文を投稿から抽出する．この抽出文はテスト時には要約として使用される．Predictor は Extractor が抽出した投稿の文を用いて，返信候補の真偽を判定する．各構成要素の詳細を以下に説明する．

■Encoder　Encoder は投稿の特徴量を計算する．まず，投稿を N 個の文 $\{s_1^p, s_2^p, ..., s_N^p\}$ に分割する． 各文 s_i^p $(i \in \{1, ..., N\})$ は K_i 個の単語 $W_i^p = \{w_{i1}^p, w_{i2}^p, ..., w_{iK_i}^p\}$ を含む．単語は単語埋め込み層を通じて，連続空間のベクトル $X_i^p = \{\boldsymbol{x}_{i1}^p, \boldsymbol{x}_{i2}^p, ..., \boldsymbol{x}_{iK_i}^p\}$ に埋め込まれる．各文の特徴量 $\boldsymbol{h}_i^p$ は各埋め込み単語ベクトルを双方向 Long Short-Term Memory (BiLSTM) に入力し，最初と最後の隠れ層を結合することで得る．

$$\boldsymbol{h}_i^p = \mathrm{BiLSTM}(X_i^p) \tag{3.1}$$

■Extractor　Extractor は Attention 機構を用いて返信候補の真偽判定に利用する少数の文を投稿から抽出する．IQE が正確な判定を行うようパラメータを学習する過程で，Extractor は返信が言及しやすい文を抽出するように学習する．これは，返信に言及されない文は返信候補の真偽判定に有用では無いからである．Extractor は抽出文の選択に返信の特徴量を使わないことに注意されたい．これは，要約の抽出処理を返信に依存させないようにするためである．IQE は学習時にのみ返信を必要とし，評価時には返信なしで要約を抽出することができる．

我々は Extractor における特徴量計算器として LSTM を使用する．Extractor の隠れ層の初期値 $\boldsymbol{h}_0^{ext}$ として，Encoder の隠れ層 $\boldsymbol{h}_i^p$ の平均値を使用する．Extractor の隠れ層 $\boldsymbol{h}_t^{ext}$ は

入力ベクトル $\boldsymbol{x}_{t-1}^{ext}$ を逐次的に入力することで更新される．この逐次的な更新は，抽出したい文の数 L 回分行われる．

$$\boldsymbol{h}_t^{ext} = \begin{cases} \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{h}_i^p & (t=0) \\ \mathrm{LSTM}(\boldsymbol{x}_{t-1}^{ext}) & (1 \leq t \leq L) \end{cases} \tag{3.2}$$

Extractor の LSTM への入力ベクトル $\boldsymbol{x}_t^{ext}(0 \leq t \leq L)$ は，以下のように計算される．

$$\boldsymbol{x}_t^{ext} = \begin{cases} \boldsymbol{p} & (t=0) \\ \sum_{i=1}^{N} \alpha_{ti} \boldsymbol{h}_i^p & (1 \leq t \leq L) \end{cases} \tag{3.3}$$

$t=0$ の時，$\boldsymbol{x}_t^{ext}$ はパラメータベクトル $\boldsymbol{p}$ であるが，$t>0$ の時，Encoder で計算された投稿文の特徴量ベクトル $\boldsymbol{h}_i^p$ の線形和である．この線形和は，Attention 機構によって計算される．図 3.2 に示すように Extractor は LSTM と Attention 機構を用いて逐次的に文を抽出する．図の $\boldsymbol{h}_k^p$ および，$\boldsymbol{h}_j^p$ は抽出された投稿の文を示し，k，および j は，1 から N の整数である．Attention 機構を用いて抽出する投稿文を一意に決定するには，Attention の重みベクトルを one-hot にする必要がある．このため，学習時には Gumbel Softmax (Jang et al. 2017) を使用する．one-hot 化された Attention の重み $\boldsymbol{\alpha}_t = \{\alpha_{ti}\}$ は，以下の式によって計算される．ここで，t は 1 から L までの整数である．

$$u_i \sim \mathrm{Uniform}(0,1) \tag{3.4}$$

$$g_i = -\log(-\log u_i) \tag{3.5}$$

$$a_{ti} = \boldsymbol{c}^T \tanh(\boldsymbol{h}_t^{ext} + \boldsymbol{h}_i^p) \tag{3.6}$$

$$\pi_{ti} = \frac{\exp a_{ti}}{\sum_{k=1}^{N} \exp a_{tk}} \tag{3.7}$$

$$\alpha_{ti} = \frac{\exp((\log \pi_{ti} + g_i)/\tau)}{\sum_{k=1}^{N} \exp((\log \pi_{tk} + g_k)/\tau)} \tag{3.8}$$

式 (3.4)(3.5) のように，Gumbel ノイズ $\boldsymbol{g} = \{g_i\}$ を，一様分布からサンプリングされたノイズ $\boldsymbol{u} = \{u_i\}$ を用いて生成する．次に式 (3.6) のように，重み a_{ti} を，Extractor の隠れ層 $\boldsymbol{h}_t^{ext}$ と Encoder が計算した各投稿文の特徴量ベクトル $\boldsymbol{h}_i^p$ から計算する．重みベクトル $\boldsymbol{a}_t = \{a_{ti}\}$ は，式 (3.7)(3.8) を通じて，one-hot に近い重みベクトル $\boldsymbol{\alpha}_t = \{\alpha_{ti}\}$ に変換される．$\boldsymbol{c}$ はパラメータベクトルであり，温度定数 τ は 0.1 に固定した．

■Predictor　抽出された投稿の文と返信候補の特徴量を使い，Predictor は返信候補が実際の返信であるかどうかを判定する．実際の投稿に正のラベル，ランダムにサンプリングされた返

信候補に負のラベルを付与した．返信候補 $R = \{s_1^r, s_2^r, ..., s_M^r\}$ は M 個の文から成ると仮定する．これらの文の特徴量ベクトル $\boldsymbol{h}_j^r$ $(j \in \{1, ..., M\})$ は，数式 (3.1) と同様に計算される．

投稿と返信候補の関係の計算に，Decomposable Attention (Parikh et al. 2016) を使用する．Decomposable Attention は，2 つのテキストを文ごとに分解し，テキスト間の関係性予測に使用するモデルである．IQE も同様に，投稿を文単位に分割して 2 つの文集合の関係性を計算するため，同様の機構を持つモデルとして，Decomposable Attention を採用した．抽出された文のベクトル $\boldsymbol{x}_t^{ext}$，および返信の文のベクトル $\boldsymbol{h}_j^r$ を Decomposable Attention に入力し，得られた出力をシグモイド関数に入力することで，二値分類の確率値 y を得る．

$$y = \text{sigmoid (Decomposable Attention}(\boldsymbol{x}_1^{ext}, ..., \boldsymbol{x}_L^{ext}, \boldsymbol{h}_1^r, ..., \boldsymbol{h}_M^r)) \tag{3.9}$$

この分類タスクの損失関数 $\mathcal{L}_{rep}$ は，以下のように Cross Entropy を用いて計算される．t_{rep} は返信候補が実際の返信である時 1 であり，そうでない時には 0 である．

$$\mathcal{L}_{rep} = -t_{rep} \log y - (1 - t_{rep}) \log (1 - y) \tag{3.10}$$

3.3 実験

我々は，2 種類のデータセットで学習と評価を行う．1 つはメールデータであり，もう 1 つはソーシャルメディア Reddit のデータセットである．

3.3.1 メールデータセット

我々は Avocado collection[*1]を学習に使用する．Avocado collection は倒産した IT 企業から得た 279 のアカウントの公開メールデータセットである．このデータセットから，投稿と返信の対を収集し，モデルの学習データを作った．

投稿と返信の対の内，単語数が 50 語以下の投稿，あるいは 25 語以下の返信を含む対を除去した．この前処理の後，56,174 の対を得た．ランダムサンプリングにより，誤った投稿と返信の対を取得する．実際の投稿と返信の対に正のラベル，誤った投稿と返信の対に負のラベルを付与した．正のラベル，負のラベルの数は同数である．すなわち，合計して，112,348 の対を得た．

評価のため，Enron 要約データセットを使用する (Loza et al. 2014)．このデータセットは 2 つの評価データセットから成る．ECS (Enron Corporate Single) と EPS (Enron Personal Single) である．ECS は業務用メールのデータセットであり，EPS は私用メールのデータセッ

[*1] https://catalog.ldc.upenn.edu/LDC2015T03

トである．各メールの要約は 2 名のアノテーターによって，2 つずつ作成されている．表 3.1 に，これらのデータセットの概要を示す．これらのデータセットは開発データセットを含まないため，ECS の開発データセットを EPS，EPS の開発データセットを ECS とした．開発データセットは，評価に使うモデルの決定に使用する．

データ	サンプル数	要約			本文		
		参照要約数	平均文数	平均単語数	平均文数	平均単語数	文辺り平均単語数
ECS	109	2	4.7	78.0	11.0	179.4	16.3
EPS	103	2	5.8	88.0	19.3	217.1	11.2
tldr	1260	1	1.6	29.7	16.6	356.6	22.4

表 3.1　評価データセットの概要.

3.3.2　Reddit TIFU データセット

Reddit TIFU データセット (Kim et al. 2019) は，tldr タグを要約タスクに応用したデータセットである．tldr は“too long didnt read”の略である．Reddit の TIFU と呼ばれるオンライン掲示板では，本文が長すぎる時に要約を tldr として記す慣習がある．我々はメールデータセットと同様の前処理を TIFU データセットに対しても行う．

既存研究 (Kim et al. 2019) で公開された TIFU データセットには返信データが付いていないため，我々は praw[*2]を用いてこれを収集した．結果，183,500 の投稿と返信の対を得た．ランダムサンプリングにより，負のラベルの付いた対を同数得て，学習データとして合計 367,000 対を得た．学習データに含まれていない投稿と tldr の対の内，tldr の単語長が 20 以上のものを 2,500 対得て，1,240 を開発データセット，もう 1,260 を評価データセットとした．TIFU の評価データセットの概要をメールデータセットと同じく表 3.1 に記す.

3.3.3　学習

単語埋め込み層と LSTM の隠れ層の次元はそれぞれ 100 とした．単語数は 30,000 とした．メールやソーシャルメディアの投稿と返信を文に分割した後に単語に分割した．分割には nltk の tokenizer[*3]を用いた．文数の上限を 30 文，各文の単語数の上限を 200 語とした．学習の epoch 数を 10 とし，ミニバッチサイズは 64，optimizer として Adam (Kingma and Ba 2015) を用いた．Adam のパラメータとして，$\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 10^{-8}$ を用いた．

[*2] https://praw.readthedocs.io/
[*3] https://www.nltk.org

最初の数 epoch では Extractor を使用せずに，投稿の全ての文を返信候補の真偽判定に使用する．これは Extractor と Predictor をそれぞれ効率的に学習するためである．Extractor はどの文を抽出すべきかを学習するが，Predictor は投稿と返信候補の関係を学習する．そのため，2 つの構成要素が別々のことを同時に学習しなければならない．一般に，複数の構成要素を持つモデルは，各構成要素を事前学習する方が高性能になる (Hashimoto et al. 2017)．そのため，最初の数 epoch では Predictor のみを学習し，Extractor を学習しない．この閾値は，全投稿文を返信候補の真偽判定に利用した時の分類精度 (F1-score) が収束する epoch 数から決定する．

学習時には，Extractor が抽出する文数 L を 1 から 4 にランダムに変化させた．投稿あるいは返信候補のテキスト中に出現する固有表現 (人物，場所，組織) を，Stanford Named Entity Recognizer (NER)[*4]を用いてタグに置換した．これは，モデルがただ固有表現のみを抽出の判断材料に使うことを防ぐためである．単語埋め込み層のベクトルは Skipgram を用いて事前学習したものを初期値とした．その際，学習には IQE 自体の学習データと同じデータを使用した．我々は同じ実験を 5 回繰り返し，平均値を結果として計算する．これは，学習パラメータの初期値や最適化の際の乱数の影響を小さくするためである．

3.3.4 評価

評価時には，Encoder と Extractor のみを使用し，Predictor を使用しない．IQE とベースライン要約モデルは 3 文を要約として抽出する．先行研究に倣い，我々は ROUGE-1, ROUGE-2，ROUGE-L の F1 値のデータ毎の平均値を評価に使用する (Lin 2004)．各モデルは 3 文を抽出し，評価に使用する．ROUGE の計算には ROUGE 2.0 (Ganesan 2015) を使用する．この時，stemming，同義語処理，および stop word は使用していない．開発データの評価指標として，ROUGE-1 の F1 値を使用する．

3.3.5 ベースラインモデル

ベースラインモデルとして，TextRank (Mihalcea and Tarau 2004)， LexRank (Erkan and Radev 2004)，KL-Sum (Haghighi and Vanderwende 2009)，PacSum (idf)，PacSum (BERT) (Zheng and Lapata 2019)，Lead，そして Random を使用する．

TextRank，および LexRank はグラフ Centrality ベースの手法で，教師なし要約において長らく強力な手法として認められてきた．PacSum は TextRank を改良した手法である．こ

[*4] https://nlp.stanford.edu/software/CRF-NER.shtml

れは，Centrality に加え，文の順番を特徴量として利用する．PacSum は文の類似度グラフ計算に，古典的な単語の共起度に idf を積算したものを用いることもできるが，BERT (Devlin et al. 2019) などのニューラルネットワークの分散表現を利用しても有効性を発揮する．本実験では，前者のモデル PacSum (idf) と，後者のモデル PacSum (BERT) 両方で実験を行う．KL-Sum は抽出された文と本文が同じ単語分布を持つように KL Divergence を用いて制約をかける手法である．Lead は最初の数文を抽出する手法であるが，ニュース記事要約において強力な手法とされてきた．加えて，ベースラインとして IQETextRank を使用する．これは文同士の類似度として，IQE の Encoder の文ベクトルの Cosine 類似度を使用した TextRank のモデルである．このモデルで実験を行う理由は，我々の提案手法の性能が単にニューラルネットワーク由来でないことを示すためである．

3.4 結果と考察

表 3.2， 3.3，3.4 にそれぞれの評価データセットに対する結果を示す．表中の † は，提案手法 IQE(固有表現置換有り) が有意に勝っている箇所，♮ は，IQE が優位に劣っている箇所を示している．有意差検定には Paired Bootstrap Resampling (Koehn 2004) を使用した．各モデルの出力の評価値を評価データの総数の半分に当たる回数分，重複有りでランダムサンプリングし，提案手法の性能が上回る確率を 1000 試行行うことで計算する．我々の提案手法 IQE はメールデータセット (ECS，EPS) において，ROUGE-2-F の評価値が PacSum (BERT) に及ばなかったものの，他の多くのベースラインモデルの性能を評価指標において上回った．Reddit TIFU データセットにおいては，IQE は TextRank 以外のベースラインモデルの性能を上回った．

ECS，および EPS データセットにおいて，IQETextRank は性能面において，IQE を大きく下回った．これは，提案手法の有効性が単にニューラルネットワーク由来のものでは無いことを支持する．過去の研究 (Zheng and Lapata 2019) においても，TextRank のグラフの類似度計算にニューラルネットワークの特徴量を利用したモデルは性能が低下することが報告されている．

提案手法は ECS データセットよりも EPS データセットにおいて，より大きく LexRank や TextRank を上回った．この理由は表 3.1 に記されたデータセットの概要から説明できる．各文中の平均単語数は EPS の方が少ない．LexRank や TextRank のようなベースラインモデルは単語の共起情報を利用した文類似度を文抽出に利用する．そのため，文長が短ければ共起ネットワークは疎になり，文間の関係性をうまく捉えられなくなり，性能が低下する．

モデル	ROUGE-1-F	ROUGE-2-F	ROUGE-L-F
Lead	0.436†	0.258†	0.336†
TextRank	0.465	0.271	0.355
LexRank	0.448†	0.260†	0.337†
Random	0.381†	0.203†	0.314†
KL-sum	0.395†	0.215†	0.313†
PacSum (idf)	0.469	0.281	0.356
PacSum (BERT)	**0.478**	**0.288**♮	**0.363**
IQETextRank	0.424±0.008	0.238±0.005	0.326±0.005
IQE	0.474±0.006	0.273±0.006	0.357±0.004
IQE 固有表現置換無し	0.456±0.007	0.263±0.009	0.348±0.005

表 3.2　ECS データセットにおける結果．太字は最も良い結果を，下線は 2 番目に良い結果を示している．提案手法の ± は，5 回実験した際の標準偏差を示す．† は提案手法 IQE が有意に勝っている箇所，♮ は提案手法 IQE が優位に劣っている箇所を示す ($p < 0.05$; Paired Bootstrap Resampling)．

モデル	ROUGE-1-F	ROUGE-2-F	ROUGE-L-F
Lead	0.246†	0.112†	0.233†
TextRank	0.332†	0.159†	0.298†
LexRank	0.316†	0.143†	0.283†
Random	0.244†	0.102†	0.236†
KL-sum	0.328†	0.157†	0.284†
PacSum (idf)	0.358	0.169†	0.310
PacSum (BERT)	**0.362**	**0.185**♮	**0.316**
IQETextRank	0.325±0.008	0.154±0.007	0.292±0.007
IQE	0.360±0.008	0.178±0.008	0.312±0.009
IQE 固有表現置換無し	0.325±0.006	0.155±0.005	0.302±0.008

表 3.3　EPS データセットにおける結果．太字は最も良い結果を，下線は 2 番目に良い結果を示している．提案手法の ± は，5 回実験した際の標準偏差を示す．† は提案手法 IQE が有意に勝っている箇所，♮ は提案手法 IQE が優位に劣っている箇所を示す ($p < 0.05$; Paired Bootstrap Resampling)．

モデル	ROUGE-1-F	ROUGE-2-F	ROUGE-L-F
Lead	0.186†	0.032†	0.146†
TextRank	<u>0.210</u>	**0.050**♮	**0.169**♮
LexRank	0.205†	0.041†	0.158†
KL-Sum	0.197†	0.042†	0.155†
Random	0.203†	0.038†	0.159†
PacSum (idf)	0.203†	0.039†	0.158†
PacSum (BERT)	0.204†	0.039†	0.158†
IQETextRank	0.209±0.001	0.045±0.001	0.164±0.001
IQE	<u>0.210</u>±0.001	0.046±0.001	0.163±0.002
IQE 固有表現置換無し	**0.211**±0.001	<u>0.047</u>±0.001	<u>0.165</u>±0.001

表 3.4　TIFU tldr データセットにおける結果．太字は最も良い結果を，下線は 2 番目に良い結果を示している．提案手法の ± は，5 回実験した際の標準偏差を示す．† は提案手法 IQE が有意に勝っている箇所，♮ は提案手法 IQE が優位に劣っている箇所を示す ($p < 0.05$; Paired Bootstrap Resampling)．

Reddit TIFU データセットにおいて，LexRank や PacSum の ROUGE 値は低い結果となった．これは，idf や順序情報が Reddit TIFU データにおいて有効でないことを示唆している．また，提案手法は Reddit TIFU データセットにおいて，TextRank より低性能となった．考えられる理由として，IQE の学習が Reddit データにおいて，メールデータより難しいということが挙げられる．IQE の返信候補の真偽判定の精度 (F1-score) は，5 回平均で，メールデータでは 0.803 であったが，Reddit データでは 0.741 であった．この理由には，それぞれのデータの投稿数の長さと，返信の長さが関連している．IQE は，投稿から一部の文のみを，返信候補の真偽判定に使用する．そのため，投稿が長ければ長いほど，返信が言及しやすい文を抽出する難易度が増加する．反対に，返信は長ければ長いほど判断材料が増えるため，抽出した文が返信と関係しているかの判断が容易になる．学習データにおけるメールデータの投稿の平均文数，返信の平均単語長はそれぞれ 7.7，150.0 であるが，Reddit データの投稿の文数，返信の平均単語長はそれぞれ，16.9，101.5 であった．このことが Reddit データにおける学習の難しさを示している．返信の長さに差が生じている理由は，Reddit は誰でも返信ができるという特性があり，それにより，メールと比べてユーザーが重要箇所に返信しなければならないという義務感を持ちにくいという点が考えられる．

3.4.1 引用抽出性能と要約性能との関係

提案モデルはメールデータセットにおいて高い性能を発揮したが，2 つの疑問が残る．1 つは，提案モデルは引用を直接教師データとはしておらず，そのため提案モデルが実際に引用箇所を抽出しやすいのか不明確であるという点である．もう 1 つの疑問は，引用と要約性能の関係についてである．我々は Carenini らの先行研究 (Carenini et al. 2007; Oya and Carenini 2014) に従い，引用が要約に有用だと仮定した．しかし，この先行研究では引用を補助的な特徴量として使用しているので，引用自体が要約に有用なのか明らかでない．これら 2 つの疑問を明らかにするために，2 つの実験を行った．

3.3.1 節で使用した Enron データセット (Loza et al. 2014) には引用を含む返信がほとんど存在しない．そのため，実験において我々は Reddit TIFU データセットと praw を通じて得た返信を利用する．このデータセットは 3.3.2 節で説明したものであり，ここから引用を含む返信を抽出する．ここで，“>”という記号から始まり，かつ返信先の投稿に含まれている文を引用と定義する．合計して，1,969 の投稿と引用を含む返信の対を得た．返信で引用される文を，要約モデルがどの程度正確に投稿から抽出できるか，また，引用が要約として有用でありうるかを，これらのデータを用いて評価する．比較には，表 3.4 において，最も良い結果を出した TextRank を用いる．

■提案モデルの引用抽出性能　要約モデルの引用抽出能力を評価するため，引用抽出を情報抽出タスクとみなし，Precision@3 (prec@3) で評価した．これは，各要約モデルが抽出した 3 つの文の中にどの程度引用文が含まれているかを示す．表 3.5 に結果を示す．IQE は TextRank，そして Random より引用を抽出する性能が高いことを示している．

■被引用箇所は要約になりうるか？　引用が要約として有用であるかを検証するため，我々は引用自体を要約とみなし，その ROUGE 値を計算した．我々は前項で説明した Reddit から得た引用データ 1,969 を引用の文数に応じて分類し，それぞれ ROUGE 値を計算した．また比較のため，TextRank やランダムに同数の文を取得した場合 (Random) の ROUGE 値も

モデル	prec@3
TextRank	0.051
Random	0.039
IQE	**0.062**

表 3.5　各モデルの引用抽出性能.

計算した．表 3.6 に結果を示す．引用文の文数が 1 文あるいは 2 文の時，引用 (Quote) は，TextRank の性能を ROUGE-2-F，ROUGE-L-F において上回った．この結果は，引用が要約として有効であるという仮説を支持する．引用が 3 文である時，性能は著しく減少した．これは引用が基本的に連続した文であるという制約があるのに対し，TextRank は文の順序を問わず，最適な文を自由に抽出できるからと考えられる．

Model	ROUGE-1-F			ROUGE-2-F			ROUGE-L-F		
	抽出文数／引用の文数			抽出文数／引用の文数			抽出文数／引用の文数		
	1	2	3	1	2	3	1	2	3
TextRank	**0.174**	**0.171**	**0.159**	0.032	0.031	**0.029**	0.123	0.132	**0.135**
Random	0.116	0.152	0.154	0.015	0.019	0.023	0.091	0.122	0.130
Quote	0.162	0.167	0.146	**0.048**	**0.044**	0.020	**0.126**	**0.136**	0.125

表 3.6　引用を抽出要約とみなした時の ROUGE. 太字は最も良い結果を示している．

3.4.2　Ablation Tests

■固有表現置換の影響　固有表現の影響を調べるため，固有表現置換を行わない場合の結果を議論する．表 3.7 のメールデータにおける結果は，固有表現置換は性能向上に寄与することを示している．これは，人名，地名，組織名は正しい返信を判定する際の大きなヒントになり得るからと考えられる．例えば，投稿と返信が同じ人物の名前に言及していると，モデルは人物名を含む文を単純に抽出するようになる．固有表現の置換は，固有表現ではなく意味的に返信に関連する文をモデルに抽出させることを促す．

しかしながら，表 3.7 が示すように，Reddit TIFU データセットでは固有表現置換はあまり性能に影響を与えなかった．Reddit は匿名掲示板であるため，投稿は特定の人物名に言及しにくいと考えられる．そのため，固有表現が投稿と返信候補の真偽判定の手がかりにはなりにくい．

■事前学習の影響　3.3.3 節で説明したように，我々は Predictor を最初の数 epoch で事前学習した．これは Extractor と Predictor を別々に学習するためである．表 3.7 に Predictor の事前学習の影響の結果を示す．事前学習なしでは，性能は低下した．これは各構成要素を別々に学習させることの重要性を示している．

■Gumbel Softmax の影響　提案モデル IQE の Extractor は本文から文を抽出するという動作を Gumbel Softmax を用いて実現している．Gumbel Softmax を使用しない場合の結果を，表 3.7 に示す．IQE の要約性能は Gumbel Softmax を用いる場合に比べて大きく低下した．

データ	モデル	ROUGE-1-F	ROUGE-2-F	ROUGE-L-F
ECS	IQE	0.474	0.273	0.357
	IQE - 固有表現置換	0.456†	0.263†	0.348
	IQE - 事前学習	0.436†	0.246†	0.338†
	IQE - Gumbel Softmax 無し	0.428†	0.240†	0.330†
EPS	IQE	0.360	0.178	0.312
	IQE - 固有表現置換	0.325†	0.155†	0.302
	IQE - 事前学習	0.322†	0.149†	0.292†
	IQE - Gumbel Softmax 無し	0.316†	0.148†	0.289†
TIFU	IQE	0.210	0.046	0.163
	IQE - 固有表現置換	0.211	0.047	0.165
	IQE - 事前学習	0.204†	0.040†	0.158†
	IQE - Gumbel Softmax 無し	0.202†	0.042†	0.157†

表 3.7　Ablation test の結果．固有表現置換や事前学習を行わない場合の結果を示す．† は IQE が有意に勝っている箇所を示す ($p < 0.05$; Paired Bootstrap Resampling)．

これは，Gumbel Softmax を用いない場合は，本文の多くの文の特徴量を返信候補の真偽判定予測に使用することができるため，Extractor が上手く学習されないことに起因すると考えられる．

3.4.3　既存手法との相違点

3.1 節で述べたように，ほとんどの既存の教師なし要約手法は重要なトピックが高頻度に言及されるという仮定に基づいている．TextRank はその好例である．TextRank は文の類似度グラフの Centrality を重要文抽出に使用する．多くの場合 Centrality の指標として PageRank が使われる．すなわち，TextRank は PageRank の高い文を要約として抽出する．ある文の PageRank が高いということは，その文が他の多くの文と類似度が高いことを意味している．これはその文が言及する話題に他の文も言及していることを示す．我々は，重要なトピックは必ずしも頻繁に言及されるわけではないと考え，言及頻度とは異なる性質に着目した．それは返信による引用のされやすさ，である．

TextRank と比較し，我々の提案手法が Centrality ベースの手法では捉えきれない重要文を抽出できることを示す．図 3.3 に最大 PageRank と各モデルが抽出する文の ROUGE-1-F の関係性を示す．各テキストを文に分割し，文毎に PageRank を計算する．その内，最大の PageRank を小数点第一位で四捨五入し，同一の最大 PageRank 値を持つデータ間の

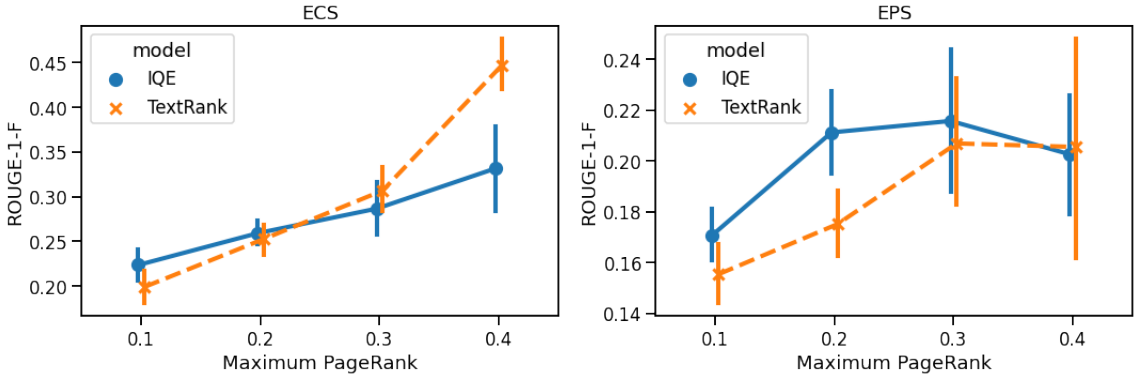

図 3.3　ECS および EPS データセットの各メールにおいて, 最大の PageRank と ROUGE-1-F の相関を示した図．X 軸は，四捨五入された最大 PageRank を，Y 軸は ROUGE-1-F を，エラーバーは標準誤差を示している．

ROUGE-1-F の標準誤差をエラーバーとして示す．使用したデータは ECS と EPS であり，ROUGE 値は各モデルが 1 文要約として抽出した時の値を示している．図より，TextRank が抽出する文の ROUGE-1-F は，最大 PageRank と比例して増加する．最大 PageRank が低い時，IQE が抽出した文の ROUGE-1-F 値は，TextRank を上回る．これは，提案手法が言及回数の少ない場合にも，重要文を捉えられるという仮説を支持している．

表 3.8 に IQE と TextRank がそれぞれ抽出した要約の具体例 (EPS データセット) を示す．参照要約は，メールの受信者の昇進や発信者の出産に言及している．これらは本文で一度しか言及されていないため，TextRank はそれらの文の抽出に失敗しているが，我々の提案モデルは正しくそれらの文を抽出できている．それは，こうしたトピックが返信によって言及されやすいからである．実際，学習に使用した Avocado Mail Collection データの返信で，baby, promoted，promotion はそれぞれ 566 回，58 回，456 回言及されている．表 3.9 に載せた Reddit TIFU データセットの具体例にも同様の傾向が見られる．すなわち，kitten や squirrel は一度しか言及されていないため，TextRank はそれらの文の抽出に失敗しているが，提案モデル IQE はそれらの文を抽出できている．

本文
Just got your email address from Rachel.
Congrats on your promotion.
I'm sure it's going to be alot different for you but it sounds like a great deal.
My hubby and' I moved out to Katy a few months ago.
I love it there - my parents live about 10 minutes away.
New news from me - I'm having a baby - due in June.
I can't even believe it myself.
The thought of me being a mother is downright scary but I figure since I'm almost 30, I probably need to start growing up.
I'm really excited though.
Rachel is coming to visit me in a couple of weeks.
You planning on coming in for any of the rodeo stuff?
You'll never guess who I got in touch with about a month ago.
It was the weirdest thing - heather evans.
I hadn't talked to her in about 10 years.
Seems like she's doing well but I can never really tell with her.
Anyway, I'll let you go.
Got ta get back to work.
Looking forward to hearing back from ya.
参照要約
The sender wants to congratulate the recipient for his/her new promotion, as well as, updating him/her about her life.
The sender just move out to Katy few months ago.
She is having a baby due in June.
She is scared of being a mother but also pretty exited about it.
Rachel is coming to visit her in couple of weeks and she is asking if he/she will join for any of the rodeo stuff.
She run into heather evans which she hadn't talked in 10 years.

表 3.8 **Implicit Quote Extractor (IQE) (bold)** と *TextRank (italic)* によって抽出された EPS データセットの文の具体例.

本文
This happened actually earlier today driving with my friend. Let me start at the beginning. *It was a sunny day and my friend and I decided to go out for some fun.* ***He came to pick me up we were about to leave the neighborhood when we feel a bump.*** Seeing as how there is no speed bump there I look back to investigate. **I saw that he had run over a squirrel but it was not dead.** Feeling remorse and wanting to put it out of its misery, we decide to back up over it to end its suffering. *As we felt the second bump, I looked up and saw the horrific look on two little girls.* Thinking the sight of a roadkill was too much for them, I casually told my friend maybe that was too graphic. **tifu: upon closer inspection, it was a kitten.**
参照要約
ran over squirrel, decided to put it out of suffering, found out it was kitten

表 3.9 **Implicit Quote Extractor (IQE) (bold)** と *TextRank (italic)* によって抽出された Reddit TIFU データセットの文の具体例.

第 4 章

要約データの適切性定量化を利用したカリキュラムラーニング

4.1 概要

ニューラルネットワークを利用した Sequence-to-sequence モデルの発展により，生成型自動要約の性能は飛躍的に向上した．Sequence-to-sequence 要約モデルの学習においては，新聞記事 (Nallapati et al. 2016) であれば見出し，ソーシャルメディア (Kim et al. 2019) やレビュー (Li et al. 2019) であればタイトル，メール (Zhang and Tetreault 2019) であれば件名を要約とみなして使用する．これらの要約は本文に書かれた内容の重要な箇所を適切かつ簡潔に記述していることが望ましい．しかしながら，過去の多くの研究が要約モデルの学習データセットには不適切な本文-要約ペアが多く含まれることを報告している (Zhang and Tetreault 2019; Li et al. 2019; Kryscinski et al. 2019; Matsumaru et al. 2020)．具体例を表 4.1 に示す．例は Reddit Title データ (Kim et al. 2019)，Enron Subject データ (Zhang and Tetreault 2019) から引用したものである．表の上段の例では本文にはタイトルの続きが書かれており，タイトルは本文に書かれている内容を反映していない．下段の例では，件名は簡潔すぎて情報不足であり，要約としての体裁を成していない．こうした要約としての品質の悪いデータは要約モデルを学習させる際のノイズとなる．こうしたノイズを含むデータセットに対処する方法が求められている．

ノイズを含むデータから効率的にモデルを学習させる方法の 1 つとしてカリキュラムラーニング (Bengio et al. 2009b) が用いられている．カリキュラムラーニングは元来，学習データの順序を変えることで，収束速度やモデルの性能を上げる手法であるが，ノイズを含むデータでモデルを学習させる際にも有効性が示されている (Wang et al. 2018, 2019; Kumar et al.

データ種類	タイトル／件名	本文
Reddit Title	accidentally drinking 0 day old coffee w / milk that was sitting on my desk next to my new coffee	just happened will update with further details as they emerge
Enron Subject	hey	here is what vickie told me about capacity on your pipeline after you eliminate segmenting . example # 1 assumtion : peco has num dt/day of telescoped capacity with a primary delivery point of peco in z6

表 4.1　タイトル，件名が本文の要約として不適切な例

2019)．しかしながら，これまでカリキュラムラーニングは要約タスクに応用されてこなかった．本章の研究の目的の 1 つはカリキュラムラーニングの要約タスクへの有効性を検証することである．

カリキュラムラーニングにおける学習データの順序の変更には，ノイズの量や難易度を表す指標が通常用いられる．学習はノイズの多いデータ群あるいは難易度の低いデータ群から始まり，徐々にノイズの少ないものあるいは難易度の高いものに移行する．ソートの際に使用する指標として，文生成タスク (Cirik et al. 2016) や翻訳タスク (Kocmi and Bojar 2017; Platanios et al. 2019; Zhou et al. 2020) においては，出力文の長さが難易度の指標として用いられている．ノイズを表す指標として，翻訳タスクにおいて 2 つの生成モデルの尤度差を用いて，カリキュラムラーニングに適用した研究がある (Wang et al. 2018, 2019; Kumar et al. 2019)．2 つの生成モデルはノイズの少ないコーパスとノイズの多いコーパスでそれぞれ学習した Sequence-to-sequence モデルである．ここではノイズは翻訳元の文章と翻訳先の文章で対応の取れない情報を指している．

要約分野においては，新聞記事などのデータセットはソーシャルメディアやメールのデータセットに比べてノイズが少ないと考えられる．しかし，要約データは要約の長さ，Density (要約箇所が本文の全体か，一部分かを示す指標) ，圧縮率，抽出率 (要約の単語が本文に含まれる割合) などの性質がデータセットによって大きく異なる (Zhong et al. 2019)．異なるデータセットで学習したモデルは，ノイズのみでなく，こうした性質を考慮したモデルになってしまう問題がある．そのため，先行研究 (Wang et al. 2018, 2019; Kumar et al. 2019) を要約モデルに適用する場合，同じドメインでノイズの多寡のみが異なるデータセットが必要になるが，こうしたデータセットは存在しない．そこで本章の研究のもう 1 つの目的として，ノイズを含

む単一コーパスからノイズを定量化してカリキュラムラーニングに適用する手法を提案する．

本章では，ノイズを含む単一コーパスからノイズを定量化できるモデル *Appropriateness Estimator* を提案する．本モデルは本文-要約の正しいペアと，ランダムに組み合わせたペアを分類する．ランダムに組み合わせたペアの要約は本文の内容を反映していない不適切なものである．不適切なペアと実際のペアを分類するように学習することで，Appropriateness Estimator は本文-要約ペアの “適切性”が判別可能になる．この適切性をカリキュラムラーニングに適用する．すなわち，適切性をデータのソートに使用し，要約モデルの学習時，学習データを不適切なペアから適切なペアへと徐々に変化させる．

本章ではノイズを多く含む要約のデータセットとして，2 つのデータセットで実験を行った．Enron Subject データセット (Zhang and Tetreault 2019) と Reddit Title データセット (Kim et al. 2019) である．両者とも学習データにはノイズが多く含まれるが，Enron Subject データセットの開発データセットと評価データセットは，人手により整理されたものである．一方 Reddit Title データセットの開発データセット，評価データセットはノイズを含む生のデータセットである．

本章では，要約タスクに対するカリキュラムラーニングの有効性と，提案手法の効果を検証するため，3 つの要約モデルと 3 つのカリキュラムで実験を行う．要約モデルには，事前学習要約モデルと非事前学習要約モデルを用いる．事前学習モデルとして BART (Lewis et al. 2020), 非事前学習モデルとして Transformer (Vaswani et al. 2017) と Seq2seqWithAttention (Bahdanau et al. 2015) を採用する．実験において，カリキュラムラーニングおよび提案手法である Appropriateness Estimator は事前学習モデル，および非事前学習モデル両方の性能を改善した．

カリキュラムラーニングに用いられるカリキュラムにはいくつかの種類が存在する．学習データを徐々に変更するもの，学習データを徐々に増やしていくもの，学習データを徐々に減らしていくものなどがある．実験結果から，事前学習モデルに有効なカリキュラムと非事前学習モデルに有効なカリキュラムが異なることが判明した．事前学習モデルにとっては，終盤に少数のデータで Fine-tuning を行うカリキュラムが有効であり，非事前学習モデルにとっては序盤に多数のデータで汎化を行うことが有効であった．また，人手による評価を行い，提案手法である Appropriateness Estimator をカリキュラムラーニングに適用した方法が要約モデルの性能を向上させることを示した．

要約のデータの性質の評価に，抽出率 (要約の単語が本文に含まれる割合) (Kim et al. 2019) や，含意判定確率 (Matsumaru et al. 2020) がこれまで用いられてきた．本章で提案した適切性をこれらの性質や入力長，出力長などの統計量と比較し，適切性の性質を議論する．加えて

これまでカリキュラムラーニングに用いられてこなかった上記抽出率や含意判定確率が要約タスクにおけるカリキュラムラーニングに対して有効であることを示す．本章の研究の貢献は以下である．

- 3 つの要約モデルでカリキュラムラーニングの実験を行い，カリキュラムラーニングの要約タスクに対する有効性を示した．
- 単一のノイズを含む学習データから学習可能な，入力文と出力文の適切性を計算するモデル Appropriateness Estimator を提案し，実験により要約モデルの性能を向上させることを確認した．
- 異なるカリキュラムが事前学習モデル，非事前学習モデルの性能にどのような影響を与えるかを分析した．

4.2 手法

本節ではまずカリキュラムラーニングの説明を行い，その後，提案手法である Appropriateness Estimator の説明を行う．カリキュラムラーニングは，学習データをある指標に基づいてソートして学習する手法であるが，その指標には難易度を表す指標やノイズ量を表す指標が使われる．提案手法である Appropriateness Estimator は後者のノイズ量計算のために使われる．

■カリキュラムラーニング　カリキュラムラーニングの概要を図 4.1 に示す．カリキュラムラーニングではまず，ある指標 (e.g. ノイズの少なさ，出力長など) に基づいて学習データを昇順にソートする．次に，学習データをセグメントごとに分割する．分割したセグメントを使ってどのように学習を進めるかに応じて複数のカリキュラムが提案されている．

One-Pass カリキュラム (Cirik et al. 2016) は最も簡単な，あるいは最もノイズが多いセグメントから学習を開始し，モデルが収束すると，学習データとして次のセグメントのデータを使用する．Baby step カリキュラム (Cirik et al. 2016) は最も簡単な，あるいは最もノイズが多いセグメントから学習を開始し，徐々に学習データを増やしていく．これら 2 つのカリキュラムは少量のデータから学習を開始する．そのため，過学習を引き起こすリスクが存在する．過学習への対処のため，最初に全てのデータで学習を行い，徐々にデータを減らしていくカリキュラムでも実験を行う．これを Noise-Annealing カリキュラムと呼称する．いずれのカリキュラムにおいても，セグメント単位での学習を終える度，モデルのパラメータを保存し，開発データでの評価値が下がった場合には，最後に保存したパラメータに戻した後，次のセグメ

ントでの学習を開始する．

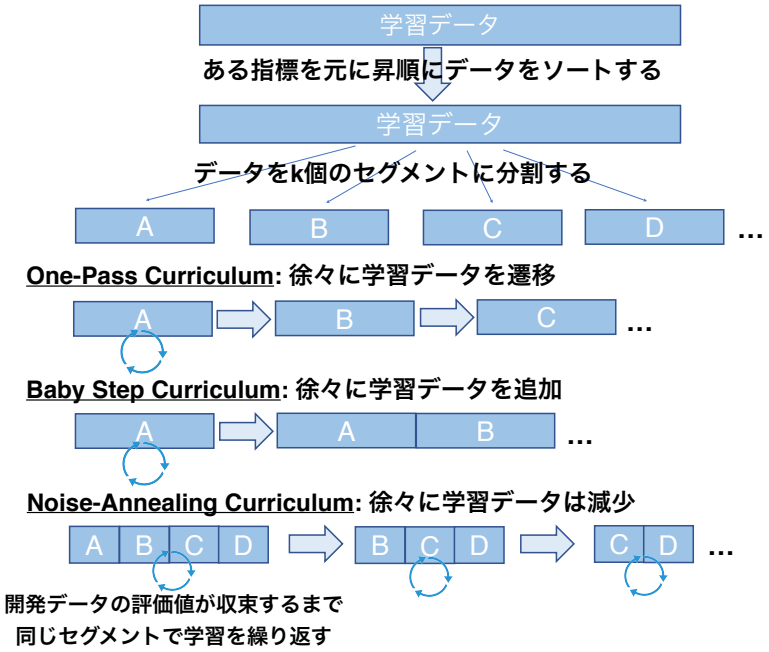

図 4.1　カリキュラムラーニングの概要.

■Appropriateness Estimator　本章では Appropriateness Estimator を提案する．本モデルはノイズを含む単一コーパスから学習可能な，ノイズを定量化するモデルである．図 4.2 にモデルの概要図を載せる．4.1 節で述べたように，要約モデルの学習データには要約として不適切なものが含まれる．ここで言う不適切とは，本文から推測が難しい情報や本文と無関係な情報が含まれているものや，記述が漠然としていて情報量に欠けているものを指す．提案モデルは本文と要約が実際の対であるかを学習することによって，本文と要約の適切性を定量化する．要約モデルの学習データに存在する本文 s_i と要約 t_k のペアを正例とし，ランダムにサンプリングされた本文 s_i と要約 t_k のペアを負例とする．ラベルを c とし，ラベルが正の時 $c = 1$ であり，負の時 $c = 0$ とする．本モデルの学習タスクは本文-要約ペアのラベル c の正負を予測することである．要約モデルの学習データに含まれるペアは全て正例とみなすが，4.1 節で説明したように，それらは不適切なペアを含む． Early Stopping は一般に過学習を防ぐ目的で使われている手法であるが，ノイズデータに対する過学習を防ぐことにも有効であることが示されている (Li et al. 2020)．本研究でも，同様に Early Stopping を利用することでモデルがノイズデータに過学習することを防ぐ．モデルの出力確率を $p(c|s_i, t_k)$ とし，損失関数 L は以下のようにクロスエントロピーを用いて計算する．

$$\mathcal{L}_{rep} = -c \log p(c|s_i, t_k) - (1 - c) \log (1 - p(c|s_i, t_k)) \tag{4.1}$$

学習後のモデルの出力確率 $p(c = 1|s_i, t_k)$ をペアの適切性と呼称する．ここで，$p(c = 1|s_i, t_k)$ が高いペアは低ノイズであり $p(c = 1|s_i, t_k)$ が低いペアは高ノイズであることを仮定している． 要約の学習データ全てを Appropriateness Estimator に入力し，適切性を計算する．ここで $c = 0$ のデータ，すなわちネガティブサンプリングにより得られたペアは含めず，元々の要約データに存在するペアのみを利用する．適切性に基づき学習データをソートし，カリキュラムラーニングを適用する．

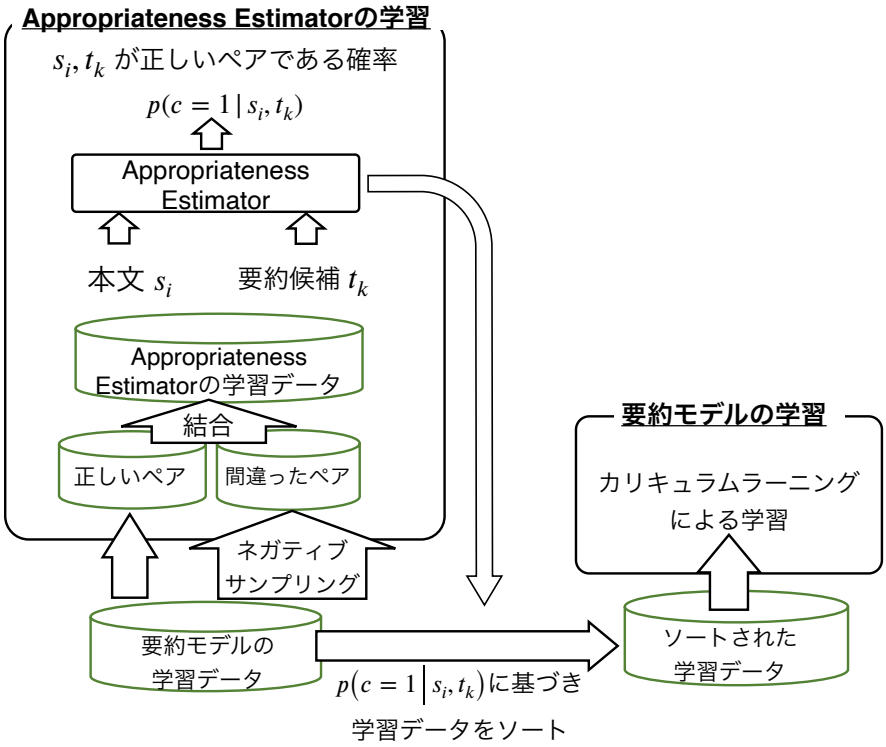

図 4.2　Appropriateness Estimator の概要図と要約モデル学習への適用.

4.3　実験

4.3.1　データセット

■Enron Subject データセット　Enron データセット (Klimt and Yang 2004) は Enron Corporation の従業員のメールを集めたデータセットである． Zhang et al. (2019) はこれらのデータを件名生成タスク用に整理した．元のデータセットは評価に使用するにはノイズが多かったため，彼らは適切な件名を人手で再度アノテーションし，開発データセットと評価データセットを構築した．学習データとして 14,436 の件名とメールの本文を使用する．開発デー

タセットとして 1,906 件，評価データセットとして 1,960 件のデータを使用する．

■Reddit Title データセット　Reddit Title データセット (Kim et al. 2019) はソーシャルメディアである Reddit のデータセットである．データは，Reddit の TIFU と呼ばれる掲示板 (Subreddit) から収集されたものである．TIFU は “today i f*** up”の略であり，投稿される文書は投稿者の失敗経験についてのものである．各投稿のタイトルを要約とみなし，要約の学習データとして使用する．学習データとして 71,113 件，開発データおよび評価データセットとして，各 3,951 件のデータを使用する．

4.3.2 モデル

■Appropriateness Estimator　Appropriateness Estimator として Decomposable Attention (Parikh et al. 2016) を利用する．単語 Embedding の初期値として GloVe[*1]を使用する．Embedding と隠れ層の次元はそれぞれ 300 と 200 とする．学習に使うエポック 数は 20 とする．

4.1 節と 4.2 節で説明したように，ランダムに割り当てた本文と出力文のペアを負例とし，元々の要約データセットに含まれるペアを正例とした．負例の数は正例の数と同数である．すなわち，Appropriateness Estimator の学習データ，開発データの数は要約モデルのそれぞれのデータ数の 2 倍となる．開発データセットにおいて最も F1-Score が高くなるエポック のモデルを要約モデルのカリキュラムラーニングに使用した．開発データセットにおける，最も高い F1-Score は Enron データセットで 0.94，Reddit データセットで 0.92 であった．

■要約モデル　本章では 3 つの要約モデルで実験を行った．その内 1 つは事前学習モデルである BART (Lewis et al. 2020) である．残りの 2 つは，非事前学習モデルである Transformer (Vaswani et al. 2017) と Seq2seqwithAttention (Rush et al. 2015) である．

Seq2seqwithAttention および，Transformer の隠れ層の次元は共に 256 とする．単語 Embedding の次元は Seq2seqwithAttention が 300，Transformer が 256 である．Seq2seqwithAttention の単語 Embedding の初期値には Appropriateness Estimator と同様に GloVe を使用する．ミニバッチ数は 3 つの要約モデルで共に 64 とする．Beam Search のサイズは 8 とする．Seq2seqwithAttention と Transformer の最適化に Adam を使用した．学習率は 0.0007 とした．BART の最適化には AdamW (Loshchilov and Hutter 2019) を使用した．学習率は 3e-5，β_1 は 0.9，β_2 は 0.999，eps は 1e-8 とした．

[*1] https://nlp.stanford.edu/projects/glove/

ランダムシードを変えつつ，同じ実験を 5 回繰り返し，平均値を結果に使用する．評価には ROUGE の F1 値 (Lin 2004) を使用する．各セグメントのエポック 毎に Validation を行い，Validation の評価指標には ROUGE-1 の F1 値を使用する．

4.3.3 カリキュラムラーニング

■カリキュラム 本章では 4 つの条件下で実験を行う．4.2 節で説明した 3 つのカリキュラムラーニングを使ったものと，カリキュラムラーニングを使わないものである．セグメント数として，5 と 10 で実験を行った．学習データの各セグメント内のデータはシャッフルする．

■データのソートに使用する指標 学習データをソートする指標として以下の 2 つを使用する．提案指標である適切性と，出力長である．出力長は難易度を表す指標として，カリキュラムラーニングで一般的に使われる指標である (Cirik et al. 2016; Platanios et al. 2019; Wang et al. 2019; Zhou et al. 2020)．本研究では，出力長は生成対象の要約文の単語数を指す．

4.4 結果

本節では，4.1 節で述べた本研究の目的であるカリキュラムラーニングの要約タスクへの有効性と，提案手法である Appropriateness Estimator の有効性の検証結果について述べる．

4.4.1 カリキュラムラーニングの要約タスクへの有効性

■自動評価 表 4.2 に，3 種のカリキュラムラーニングを要約モデルの学習に用いた場合と，用いなかった場合の実験結果を載せる．評価は自動評価指標である ROUGE を用いて行った．各モデル各データの最も高い性能 (モデル 1) は太字で記されている．† は同モデル同データ内で 2 番目に性能が高いもの (モデル 2) との有意差を示している ($p < 0.05$)．有意差検定には，Bootstrap Resampling (Koehn 2004) を使用した．テスト時のモデルの出力の件数の半分に当たる回数，ランダムに結果を重複ありでサンプリングし，モデル 1 とモデル 2 の結果を比較する．これを 1000 回繰り返し，950 回以上モデル 1 がモデル 2 の性能を上回った場合，有意とみなす．いずれのモデル，データにおいても，カリキュラムラーニングを使ったモデルが最も高い性能を示しており，要約タスクにもカリキュラムラーニングが有効であることが確認された．

■人手評価 本章では 2 つの BART モデルで人手評価を行った．1 つは Noise-Annealing カリキュラムおよび適切性で学習されたものであり，もう 1 つはカリキュラムラーニングなしで

			Reddit Title			Enron Subject		
要約モデル	カリキュラム	ソート指標	R-1-F	R-2-F	R-L-F	R-1-F	R-2-F	R-L-F
BART	無し	-	0.254	0.124	0.222	0.301	0.153	0.255
	One-Pass	適切性	0.276	**0.137**	0.243	**0.339**†	**0.193**†	**0.294**†
		出力長	0.268	0.123	0.235	0.329	0.186	0.286
	Baby step	適切性	0.230	0.108	0.200	0.277	0.136	0.236
		出力長	0.244	0.117	0.214	0.300	0.156	0.257
	Noise-Annealing	適切性	0.271	0.132	0.239	0.315	0.167	0.270
		出力長	**0.277**	0.135	**0.245**	0.312	0.171	0.271
Transformer	無し	-	0.184	0.047	0.140	0.093	0.019	0.044
	One-Pass	適切性	0.153	0.017	0.121	0.040	0.003	0.027
		出力長	0.156	0.014	0.131	0.062	0.001	0.040
	Baby step	適切性	0.170	0.027	0.131	0.079	0.012	0.056
		出力長	0.167	0.023	0.125	0.091	0.018	**0.065**†
	Noise-Annealing	適切性	**0.192**†	**0.051**	**0.146**†	**0.106**†	**0.022**†	0.047
		出力長	0.188	0.048	0.141	0.094	0.019	0.044
Seq2seqAtt	無し	-	0.171	0.041	0.118	0.051	0.006	0.031
	One-Pass	適切性	0.161	0.018	**0.119**	0.039	0.000	0.015
		出力長	0.142	0.021	0.099	0.034	0.000	0.016
	Baby step	適切性	0.167	0.027	0.112	0.051	0.006	0.029
		出力長	0.147	0.030	0.108	0.051	0.006	0.030
	Noise-Annealing	適切性	**0.176**	0.041	0.116	**0.060**	**0.008**	**0.040**
		出力長	0.172	**0.043**	0.118	0.057	**0.008**	0.036

表 4.2　カリキュラムラーニングを使った要約モデルの実験結果. 適切性は Appropriateness Estimator の出力確率を表す．R-1-F, R-2-F, R-L-F はそれぞれ ROUGE-1, ROUGE-2, and ROUGE-L の F1 値である．各モデル各データの最も高い性能は太字で記されており，† は同モデル同データ内で 2 番目に性能が高いものとの有意差を示している ($p < 0.05$)．有意差検定には，Bootstrap Resampling (Koehn 2004) を使用した.

学習されたものである．両モデルで生成された要約が同じである場合，それを除去し，計 90 の生成要約を Enron データセット，Reddit データセットそれぞれに対して得た．アノテーターはどちらの要約が良いかを情報量，流暢性の観点から評価する．情報量は生成要約が本文の重要な情報をどの程度含んでいるかを表しており，流暢性は文法の観点から生成された文が正しいかを判断する．表 4.3 に結果を示す．表は，Noise-Annealing と適切性で学習されたモデルは情報量，流暢性両方の観点でより高い評価を得たことを示している．統計的な有意性を評価するため，“良い”と“どちらかといえば良い”の票数を統合し χ 二乗検定を行った.

	情報量		流暢性	
	Enron†	Reddit†	Enron‡	Reddit†
BART (CL あり) の方が良い	**42**	**37**	22	8
BART (CL あり) の方がどちらかといえば良い	17	25	**34**	**55**
BART (CL 無し) の方がどちらかといえば良い	14	20	25	26
BART (CL 無し) の方がが良い	17	8	9	1

表 4.3　人手評価の結果．CL ありは Noise-Annealing カリキュラム使用時の結果を指し，CL なしはカリキュラムラーニング無しの結果を指す．† and ‡ は，BART (CL あり) が“良い”と“どちらかといえば良い”の票を多く得る確率を χ 二乗検定で検定したものである (†: $p < 0.01$, ‡: $p < 0.05$).

4.4.2　Appropriateness Estimator の有効性

Appropriateness Estimator の適切性を利用したカリキュラムラーニングは，3 つのモデル全てで，カリキュラムラーニングなしのモデルと比べ，性能を向上させた．出力長を利用したカリキュラムラーニングと比較した場合，適切性を利用した場合は Enron Subject データにおいてより上げ幅が大きかった．これは，Reddit Title データセットでは，テストデータが学習データ同様未整備であるのに対し，Enron Subject データのテストデータが，人手で整備されたノイズの少ないデータであることに関係していると考えられる．適切性を用いたカリキュラムラーニングは，Noise-Annealing カリキュラムと One-Pass カリキュラムの場合，学習の終盤においてノイズの少ないデータで学習を行う．これにより，学習モデルが低ノイズの高品質なデータに Fine-tuning されているため，より低ノイズのテストデータセットにおいて性能を発揮すると考えられる．

4.5　考察

本節では，本実験で用いた 3 つのカリキュラムが与える性能への影響，翻訳タスクとの相違点，適切性の性質について議論する．

4.5.1　カリキュラムごとの相違点

今回本研究では，3 種類のカリキュラムで実験を行った．One-Pass カリキュラムおよび Noise-Annealing カリキュラムは学習の終盤で少数のデータで Fine-tuning を行い，Noise-Annealing カリキュラムは学習の初期に多様なデータでの汎化を行うという特徴がある．こ

うしたカリキュラムの特徴と，カリキュラムラーニングなしと比べた際の要約モデルの性能に与える影響のまとめを表 4.4 に載せる．One-Pass および Noise-Annealing カリキュラムは BART モデルの性能を向上させたが，Baby step カリキュラムは性能を悪化させた．考えられることは，BART は事前学習モデルであるため，改めて汎化を行う必要はなく，Fine-tuning が重要であるという点である．非事前学習モデルにおいては Noise-Annealing カリキュラムのみが要約モデルの性能を向上させた．これは，事前学習モデルとは対照的に，非事前学習モデルにとっては汎化が重要であることを示唆している．

カリキュラム	学習初期の汎化	学習終盤の Fine-tuning	事前学習モデルの性能	非事前学習モデルの性能
One-Pass	なし	あり	向上	低下
Baby step	なし	なし	低下	低下
Noise-Annealing	あり	あり	向上	向上

表 4.4　3 つのカリキュラムのカリキュラムラーニングなしと比べた場合の性能差.

4.5.2　翻訳タスクとの相違点

Baby step カリキュラムは，徐々に学習データを増やしていくカリキュラムである．過去の翻訳タスクにおける研究では出力長を同様のカリキュラムに使用し，翻訳モデルの性能を向上させた (Kocmi and Bojar 2017; Platanios et al. 2019; Zhou et al. 2020) が，本稿の実験設定では要約モデルの性能は低下した．

要約タスクと翻訳タスクの出力文の違いとして，翻訳タスクでは，出力文の質と出力文の文長に相関はないが，要約タスクではその限りではないということである．翻訳タスクにおいては，出力文は入力文の長さに比例するため，入力文が短い場合は必然的に出力文も短くなり，翻訳の正確性には関係がない．しかしながら，要約タスクにおいては，出力文と入力文は比例せず，短すぎる要約は十分な情報を含んでいないと考えられる．そのため，少数のデータで学習をはじめる Baby step カリキュラムおよび One-Pass カリキュラムを非事前学習モデルに適用した場合，低品質データに対する過学習を引き起こし，性能が低下すると考えられる．

4.5.3　適切性が高い／低い本文-要約ペアの具体例

表 4.5，4.6 に適切性が高い，あるいは低い入力文（本文）と出力文（要約）のペアの例を載せる．表にある適切性の低い例の出力文は本文にない情報を含んでいることがわかる．表 4.6 の下段の例では，適切性が低いペアにおいて，出力文の続きが入力文に書かれており，出力文

Subject	本文	適切性
TW/ Lonestar Ward and Pecos Counties interconnect bi-directional– A-release	The following is a level “A” cost estimate to make TW/ Lonestar existing interconnects bi-directional. TW/ Lonestar at Ward County (50 to 60 mmcf/d) According to Operations this is already bi-directional . The only things are required on this one is to take the flapper out of the check-valve and blow down the gas in 5.33 miles of 12”. Cost of gas loss& labor = $8,000 TW/ Lonestar at Pecos County (100 mmcf/d) A): TW/ Lonestar interconnect Scope: On this one we need a bi-directional valve skid using the existing meter run. Cost of material& labor= $ 195,000 B): Pecos Compressor Station In order to make this interconnect bi-directional we also need to make the station (two-compressor units) bi-directional. Scope: Install outlet from Lonestar I/C to inlet filter with 12” piping& valves. Unit discharge would be modified to tie in to West Texas-20” Cost for material& labor= $ 330,000. If you need more accurate costs (B -release) please let me know .	0.99
Returned mail: Host unknown (Name server: enron: host not found)	Danny, I’m resending as I had the same problem Cindy did. I’ll give you a call later today after I’ve talked to Harris to discuss the various Gallup scenarios to make sure you and I are on the same page. The plan that makes the most sense in my mind is to ram the 10,000/d project through asap, with no firm contracts to preserve our options on a NEWCO structure. We’ll simultaneously implement a new approach on San Juan fuel transport if possible and then throw the big expansion into the hopper at FERC in January as Stan suggested. I hope that timetable is doable–it all depends on	0.01
GREAT NEWS ****FERC Order on Morgan Stanley Complaint Against ISO	See below. this is one of the issues that concerned us more than price caps, because it could limit our ability to move power to other markets in the west. In addition, if you get questions from the analysts on “reregulation” or price caps it is worth pointing out that the high prices prevailing in many markets help our retail sales pitch to end use customers and create opportunities for our wholesale price risk management services ... even a $250 price cap is 5-10 times what large customers are accustomed to paying.	0.01

表 4.5　適切性が高い／低い本文－要約 (Subject) ペアの例 (Enron Subject データセット).

の情報が入力文に書かれていない．対照的に，適切性が高いペアでは出力文は本文の内容を反映している．

4.5.4　適切性の表す性質

本節では，適切性が表す性質について議論する．適切性は要約データの品質，すなわちノイズの大きさと比例すると考えられる．要約データの品質は人の主観によって判断されるため厳密な定義は困難であるが，高品質な要約データの要件としては，情報量や含意性などの観点が考えられる．情報量は本文の重要な箇所を反映している度合いを指し，含意性は要約が本文に

Title	入力文	適切性
asking if my roommate had any plans for mother's day.	yesterday, technically, i was at home making myself a nice meal because i couldn't be with my family for mother's day due to distance. as i'm preparing my dinner, my roommate came into the kitchen. thinking i would be a good roommate and strike up some passing conversation, i asked him if he had any plans for mother's day, to which he replied that his mom had died just last month. he hasn't exactly made this super well known in the house, but i had seen a fb post of his last month mentioning this. i felt like the most insensitive asshole ever and apologized as well as i could. but i'll always feel bad about that one.	0.97
Backing my e class into my wife's c class mercedes	My wife had been out of town all week at a sales conference. Our driveway makes at with one car pulling to the left into our carport and one car that pulls forward to park on a concrete slab. Initially my wife was supposed to get our kids from daycare but her flight was running late so she decided to come by the house first to pick me up so we could go out to dinner. I was finishing some work projects at home when she came running in from the airport. I didn't realize we were on the verge of not picking the kids up on time. The daycare charges something like $10 a minute if you're late and it was a friday. She was gathering some things for our toddler (you can't go out with a 3 yo unless you're prepared to bring a toy store to entertain them with). I had the bright idea that I would back out of the carport and pull up so her passenger door would be readily accessible when she came out the back door (i had been pulling out that way all week so I could pull out into the street rather than back out). In a hurry, I slammed my car in r and jammed on the gas. Boom! I hit her car just as she was coming out the door. Toddler toys go flying everywhere (mostly at my head). We didn't speak all the way to the daycare until I just started laughing hysterically. I mean really. What else could you do?	0.01
Leaving a 12-pack of beer in the bottom of a shopping cart in the grocery store parking lot.	I went back to get it 30 minutes later and it was still there :)	0.01

表 4.6　適切性が高い／低い本文-要約 (Title) ペアの例 (Reddit Title データセット).

無い情報を含んでいないことを指す．こうした観点で，適切性が要約の品質を反映できているかを議論する．比較対象として，他にノイズと相関すると考えられる 4 つの性質と比較して議論する．その内の 2 つは，入力長と出力長である．3 つ目は一致する単語の割合である．Appropriateness Estimator は，本文-要約ペアの適切さを一致する単語によって判定している可能性が高い．要約に含まれる単語が本文に含まれる割合を抽出率と定義し，抽出率と適切性との関係性について議論する．4 つ目は含意関係である．松丸ら (Matsumaru et al. 2020) は，

含意判定器を用いて要約データのフィルタリングを行った．含意判定器の確率値と適切性の違いについて議論する．上記 4 つの性質と適切性とのピアソンの相関係数を表 4.7 に載せる．

また，上記 4 つの内，入力長，抽出率，含意確率を用いてカリキュラムラーニングを行った際の結果を表 4.8 に載せる．この時，カリキュラムとして用いたのは，表 4.2 の実験で有効であった One-Pass カリキュラムと Noise-Annealing カリキュラムである．要約モデルには最も性能の高い BART を使用した．表 4.2 の実験と同様に，実験はランダムシードを変えて 5 回行った際の平均値を載せる．有意差検定は，表 4.2 の実験と同様 Bootstrap Resampling (Koehn 2004) を用いて行う ($p < 0.05$)．

データセット	出力長	入力長	抽出率	含意確率
Enron Subject	0.151	0.079	0.711	0.244
Reddit Title	0.156	0.018	0.572	0.174

表 4.7　適切性と各統計量との相関係数（ピアソン）

			Reddit Title			Enron Subject		
要約モデル	カリキュラム	ソート指標	R-1-F	R-2-F	R-L-F	R-1-F	R-2-F	R-L-F
BART	無し	-	0.254	0.124	0.222	0.301	0.153	0.255
	One-Pass	適切性	**0.276**†	**0.137**	**0.243**†	**0.339**†	**0.193**†	**0.294**
		入力長	0.241	0.112	0.210	0.315	0.169	0.269
		抽出率	0.251	0.115	0.222	0.332	0.176	0.291
		含意確率	0.243	0.114	0.210	0.330	0.181	0.283
	Noise-Annealing	適切性	0.271	0.132	0.239	0.315	0.167	0.270
		入力長	0.260	0.129	0.227	0.312	0.164	0.265
		抽出率	0.269	0.134	0.236	0.312	0.160	0.265
		含意確率	0.267	0.134	0.235	0.317	0.168	0.273

表 4.8　入力長，抽出率，含意確率をカリキュラムラーニングを適用した要約モデルの実験結果．各モデル各データの最も高い性能は太字で記されており，† は同モデル同データ内で適切性以外で最も性能が高いものとの有意差を示している（$p < 0.05$）．有意差検定には，Bootstrap Resampling (Koehn 2004) を使用した．

■適切性と入力長および出力長との関係　表 4.7 より，適切性と，入力長及び出力長との相関係数は 0.2 未満であった．これは，適切性が入力長や出力長とは異なるテキストの性質を表していることを示している． 4.5.2 節で述べたように，要約タスクにおける出力長はノイズの量と相関すると考えられる．これは，短すぎる出力は要約として十分な情報を含んでいないと考えられるからである．ただし，表 4.5 や表 4.6 で示した例のように，出力長が長くてもノイズとみなせる要約も存在する．表に挙げた例の出力は本文に無い情報を含んでおり，高品質な要

約の要件の一つである含意性を満たさない，また，表 4.8 より，入力長をカリキュラムラーニングの指標として用いた場合の性能は，他の指標と比べて悪くなった．これは入力長と要約データとしての適切さが比例しないことを示唆している．

■適切性と抽出率との関係　Appropriateness Estimator は，本文と要約のペアの正しさを学習している．そのため，要約に含まれる単語が本文に含まれるかを元に判定している可能性が高い．そこで，要約に含まれる単語が本文に含まれる割合（これを抽出率と呼称する）と適切性の関係性を議論する．抽出率と適切性のピアソン相関係数は，表 4.7 にあるように高い．これは，適切性が単語の一致率を反映していることを示す．ただし，適切性が抽出率に比べ有利な点は単語の一致だけではなく，関連する単語に着目できる点と，一般的すぎる単語の重要度を相対的に下げられる点である．Appropriateness Estimator は他の異なる要約を負例として判別するよう学習するため，要約の内容が一般的で情報量が少ない場合，他の本文にも含まれる可能性が高くなり，正しい要約と誤った要約の区別がつきにくくなる．そのため，こうした本文-要約ペアには相対的に低い適切性が与えられる．表 4.9 にある例の抽出率は高いが，書かれている内容が一般的で情報量を欠いているため，適切性が相対的に低くなっている．適切性は全体的に高い数値であるため，この数値は全体の下位 23.2 %に該当する．

抽出率をカリキュラムラーニングに適用した場合の結果を表 4.8 に載せる．抽出率は Noise-Annealing カリキュラムにおいては，適切性と同等の性能であったが，One-Pass カリキュラムにおいては適切性が有意に上回った．これは前述した点で適切性が抽出率に比べ有利であるからと考えられる．また，抽出率を使った結果は，カリキュラムラーニングを使わない場合の性能を上回っており，この実験結果はこれまでカリキュラムラーニングに用いられて来なかった抽出率が要約タスクにおいて有効であることを示している．

要約	本文	抽出率	適切性
trying new things	this happened a few months ago during soccer season. i wanted to try new things and i decided to join soccer with my friend. it was my first time playing and i was pretty uncoordinated. however i had seen people playing soccer before and how they used their heads to score off of corner kicks. i thought that was pretty cool! so in my first practice i tried to head the ball in the goal. however, the ball came a lot faster than i thought it would and it got me on the wrong part of my head. it felt real sore and i gave up on trying to head the ball in.（後略）	1	0.71

表 4.9　抽出率が高いが，適切性が相対的に低い本文－要約ペアの例 (Reddit Title データセット)．

■適切性と含意確率との関係　松丸ら (Matsumaru et al. 2020) は，含意判定器を用いて要約学習データセットから不適切な本文と要約のペアを除去した．含意確率と適切性の関係性についても本節で議論する．含意判定器には，松丸ら (Matsumaru et al. 2020) と同様に，RoBERTa (Liu et al. 2019) を Multi-Genre Natural Language Inference (MultiNLI) データセット (Williams et al. 2018) で学習させたものを使用し，本文が要約を含意する確率を計算する．なお，松丸らは，MultiNLI で学習させたモデルを，要約データセットごとに独自にアノテーションしたデータセットで更に Fine-tuning したモデルを用いているが，本研究では対象要約データセットが異なるため，MultiNLI で学習させたモデルを用い，Fine-tuning は行わない．

適切な要約の要件には，要約の内容が本文にも書かれているという含意性の他に，本文の重要な箇所が要約に反映されているべきだという情報量の観点がある．Appropriateness Estimator は一般的すぎて情報量が少ない要約には相対的に低い適切性を与える．これは要約の記述内容に情報量が少ない場合，本文に対する正しい要約であるか，他の本文に対する要約であるかを判定するのが難しくなるからである．一方含意判定器は含意性のみを判定するため，こうした例を除去することはできない．表 4.10 の上段の例では，含意確率は高くなってしまうが，要約として適切な情報が含まれているとは言い難い．一方 Appropriateness Estimator は，主語と目的語の逆転や否定肯定の反転などを区別することはできないため含意性に関しては含意判定器が高性能に行うことができる．

また，既存の含意判定器は要約データセット向けに作られたものでは無いため，要約データセットに対しては適切な含意判定ができないという問題も存在する．一般的に含意判定器は同程度の長さの文同士を比較するが，要約の場合は複数の文にまたがる情報を含意することがある．既存の含意判定学習データセット MultiNLI を用いて学習された含意判定器で含意確率が高いと判定された本文-要約のペアは，表 4.10 の中段の例にあるように，同程度の長さで要約の内容が本文に存在していることが多い．これに対し，下段の例のように複数文にまたがる内容の要約は含意判定器では含意確率が低いと判定される．含意判定器の確率値と本文の単語数 (入力長) とのピアソン相関係数は，Enron Subject データセットにおいて-0.30，Reddit Title データセットにおいて-0.42 であり，いずれも負の相関性を示していた．この相関関係は本文が長すぎる時には，既存の含意判定器が上手く機能しないという仮説を支持している．

先行研究 (Matsumaru et al. 2020) では，含意判定器を要約データセットのフィルタリングに使用したが，この時含意判定器は要約データセット向けに新たにアノテーションされたものを用いて Fine-tuning されており，正確に要約データの含意性を判定するためには要約データセットに対する含意関係のアノテーションが必要となる．提案手法である Appropriateness

要約	本文	含意確率	適切性
waking up...	i woke up this morning, went out to the barn to discover that one of my kittens had died :(then i was milking a cow and she kicked me so hard in the hand i thought she broke it, then i went to jump on my boyfriends back so he could give me a piggy back ride to the house, and smashed my bad knee into an air tank in the milk house. now i can't bend my leg. and to top everything off, as i was cutting my breakfast sausage with a fork, it slipped and poured sausage covered with syrup all over me... i should have just gone back to bed...	0.98	0.65
getting drunk and buying nascar '15	i got too drunk and bought nascar '15 on xbox 360. seriously, who the fuck does that?	0.99	0.98
trying to backflip on a trampoline.	obligatory "not actually today": this happened last summer. i (then 23) was bouncing around on a trampoline with my two brothers, then 10 and 6, and my girlfriend. my brothers asked if i knew how to backflip, and i jokingly bragged "yes" although i'd never done it before. now, i mastered (trampoline) front flips pretty quickly many years ago, but always found backflips difficult. basically, i never had the guts to go through with it and never had enough spin to go past landing on my back. not this time. i figured i'd at least make an honorable effort, and i kicked myself into a spin as i jumped. apparently, the spin was just enough to turn me upside down before i landed. as i landed on my neck, my head bent forward (chin to chest) under the weight of my body, as what seemed like strong electricity trickled down my back. that shit hurt. i then went on to lecture my brothers about what happened and how dangerous it really was. never seen them attempt a backflip since then!	0.00	0.95

表 4.10　含意確率が低い，あるいは高い本文－要約ペアの例 (Reddit Title データセット).

Estimator は，このような要約データセットのアノテーションを必要とせず，含意判定器やその学習データセットが無い言語に対しても有効である．

表 4.8 の結果は，含意確率をカリキュラムラーニングに用いた場合の性能が，Enron Subject データセット，Reddit Title データセット両方において，適切性と One-Pass カリキュラムを用いた場合の性能を有意に下回っていることを示している．これは，前述したように含意確率が情報量を考慮できない点と，既存の含意確率モデルが要約データセットに対して必ずしも有効でないことによると考えられる．一方カリキュラムラーニングを用いない場合に比べ性能は向上しており，これまでカリキュラムラーニングに用いられて来なかった含意確率という指標が，要約データセットに対して有効であることを示している．

4.5.5 どのセグメントで最も高性能となるか？

One-Pass カリキュラム，および Noise-Annealing カリキュラムは両方とも少量のデータで Fine-tuning を行う．どのセグメントで最も要約モデルの開発データの評価性能がよくなるかを調査することで，どのセグメントが最も Fine-tuning に適しているかを明らかにする．

表 4.11 に開発データセットの評価性能 (ROUGE-1-F) が最も高くなるセグメントの番号を記す．モデルには BART を使用する．表には 5 回行った実験の平均値と標準偏差を記してある．セグメント数は 10 であり，10 番目のセグメントが最も長い，あるいは最も適切性が高い．出力長を用いてカリキュラムラーニングを行った際，要約モデルはより早いセグメントで最も開発データセットにおける評価性能が高くなっているが，提案した適切性を用いた時には，より後のセグメントで最も評価性能が高くなっている．これは出力が長すぎるデータは要約モデルの Fine-tuning に適切でないことを示唆している．長過ぎる要約は，情報の取捨選択が適切に行われていないからである．一方，本稿で提案した適切性は，高いものがより Fine-tuning に適した，要約モデルの学習に適したデータになっていると考えられる．

	出力長		適切性	
	Enron	Reddit	Enron	Reddit
One-Pass	3.1± 2.3	2.9± 1.3	6.6± 3.0	5.6± 3.6
Baby step	6.5± 2.4	6.5± 2.9	7.1± 2.6	6.8± 2.9
Noise-Annealing	3.5± 0.9	4.5± 1.4	7.2± 2.4	7.4± 2.4

表 4.11　開発データセットにおける評価指標 (ROUGE-1-F) が最大になるセグメント．ランダムシードを変えた時の 5 回の実験の平均値と標準偏差を記してある．セグメントの数は 10 である．

4.5.6 適切性を用いたフィルタリングの効果

今回，適切性を用いて，要約データの品質を定量化し，カリキュラムラーニングを適用することで，要約モデルの性能を向上させた．しかし，品質を定量化した後に，モデルの性能を向上させる方法としては，他にフィルタリングや損失関数の重み付けなどがある．第 2.5 節で議論したように，閾値以下の品質のデータを学習データから除去するとモデルの性能が向上することが知られている (Csáky et al. 2019; Matsumaru et al. 2020; Zhang et al. 2020)．本節では，適切性を用いて，学習データの一部をフィルタリングし，要約モデルを学習させた時の結果を検証する．本実験のカリキュラムラーニングでは，学習データを 10 のセグメントに分

割したが，本実験では分割したセグメントの最後のセグメントのみを学習に使用する．モデルは BART を使用する．結果を表 4.12 に示す．比較のため，出力長を元に学習データを分割した場合の結果も示す．表のカリキュラムラーニングは，One-Pass カリキュラムを用いた時の結果を示している．

	Reddit Title			Enron Subject		
学習方法	R-1-F	R-2-F	R-L-F	R-1-F	R-2-F	R-L-F
フィルタリング／カリキュラム無し	0.254	0.124	0.222	0.301	0.153	0.255
適切性 + カリキュラムラーニング	**0.276**	**0.137**	**0.243**	0.339	**0.193**	0.294
適切性 + フィルタリング	0.264	0.120	0.230	**0.346**	0.192	**0.300**
出力長 + フィルタリング	0.224	0.082	0.156	0.271	0.139	0.210

表 4.12　フィルタリングを用いた結果

適切性を使用した場合と，出力長を使用した場合では，適切性を使用した場合の方がはるかに性能が高い結果となった．これは適切性の方がより的確に学習データの品質を捉えられていることを示している．

カリキュラムラーニングを使用した場合の結果と比較した場合，フィルタリングは Enron データセットにおいてカリキュラムラーニングよりモデルの性能を向上させたが，Reddit データセットにおいてはカリキュラムラーニングに劣ることが判明した．一般的にフィルタリング手法では，学習データ数の低下と品質がトレードオブの関係にある．先行研究 (Wang et al. 2018) で指摘されているように，一般的に学習データは多い方がモデルの性能は向上するため，品質の低いデータでも学習データに加えた方が良い場合があり，こうした場合はカリキュラムラーニングがフィルタリング手法よりも有効である．

今回の実験で使ったデータセットでは Enron データセットの方がデータ数が少ないのでこれには当てはまらない．考えられることとしては，Enron データはメールデータであり，定型文のような表現が多いため，学習データが少なくても十分学習可能であるが，ソーシャルメディアのデータである Reddit データセットでは多様なデータが学習に必要であるという点である．

ただし，学習データの品質と数のトレードオフの関係はどこに分岐点があるのか明らかではなく，フィルタリングとカリキュラムラーニングを比較検討した過去の研究も存在しないため，今後より詳細な検討が必要になる．

第 5 章

結論と今後の課題

本論文のテーマは高品質教師データが無い状況下の要約手法の研究であった．序論である第 1 章では，整備された教師データがあるドメインに特化されて既存の要約研究が行われてきたことを指摘し，その対策案を 2 つ提示した．一つは，教師なし学習手法，もう一つは低品質教師データから効率的に学習する手法である．第 2 章では，自動要約の既存研究全体を外観しながら，これら 2 つの対策案に関連する既存研究について議論した．第 3 章で教師なし学習の新手法について，第 4 章で低品質教師データから効率的に学習する手法について議論した．本章では，第 3 章と第 4 章で議論した研究に対して，結論と今後の課題を述べる．その後，本論文のテーマに関連して，本論文で取り上げなかった手法の紹介とその課題について述べる．最後に，本論文で取り上げたテーマ全体の今後の展望について述べる．

5.1　第 3 章で提案した教師なし学習手法の結論と課題

第 3 章では，従来手法とは異なる文の重要度の指標である “返信による引用のされやすさ”に着目した教師なし抽出型要約モデル Implicit Quote Extractor (IQE) を提案し，その有効性を 2 つのメールデータセット (ECS と EPS) と 1 つのソーシャルメディアデータセット (Reddit TIFU) を対象にした評価実験で示した．また，IQE が既存手法が抽出できない重要文を抽出可能であることを定性的定量的両方の観点で示した．

我々は，従来手法とは異なる指標に着目したが，従来手法で用いられた “言及頻度”も依然として重要であるため，これら両方の側面を考慮するモデルの考案が今後の課題となる．

また，提案モデルは，返信が本文の重要箇所に言及すると仮定したが，現実的にはそうでないケースも考えられる．学習に用いた返信候補の内，重要箇所を参照していないものを上手く篩にかけることができればより性能が向上すると考えられるが，この方法についても今後の課題とする．

本研究で提案したモデル IQE は，抽出型教師なし要約モデルであるが，近年の要約研究では，生成型教師なし要約モデルの研究が活発化している．その理由の一つが GPT-2 (Radford et al. 2018) や GPT-3 (Brown et al. 2020)，BART (Lewis et al. 2020) をはじめとした大規模事前学習生成モデルの発展である．GPT-2 は文末に TL;DR を付けて生成することで，zero-shot の要約を行うことができると主張されている．他にも，GPT-2 を用いてマスクされた文を復元するタスクで生成型要約モデルを学習した研究がある (Laban et al. 2020)．提案モデルである IQE に対しても，同様に事前学習モデルを使うことで，生成型要約モデルを学習させることができる可能性があり，これは今後の課題としたい．

5.2 第 4 章で提案した低品質教師データの効率的学習手法の結論と課題

第 4 章では，これまで要約タスクにおいて取り組まれてこなかったカリキュラムラーニングを要約タスクに適用し，その有効性を検証した．本稿では，実験設定として，3 つのカリキュラム（One-Pass，Baby step，Noise-Annealing）とカリキュラムを使わない場合で実験を行った．また，実験対象の要約モデルとして 1 種類の事前学習モデル BART と 2 種類の非事前学習モデル Transformer と Seq2seqWithAttention を用いた．結果，非事前学習モデルにおいては Noise-Annealing カリキュラムのみが要約モデルの性能を向上させた．また，事前学習モデルにおいては One-Pass カリキュラムと Noise-Annealing が性能を向上させた．これらの実験結果から，少量のデータで Fine-tuning を行うことが事前学習要約モデルに重要であり，汎化が非事前学習要約モデルにとって重要であることを結論付けた．より効率的に Fine-tuning 対象のデータを探る手法を考案することが今後の課題である．

また本稿では既存の翻訳タスクで取り組まれてきたノイズを指標にしたカリキュラムラーニングを要約タスクに適用することを試みた．要約の学習データには翻訳タスクにあるようなノイズの少ないあるいは多いコーパスが存在しないため，単一のノイズを含む学習データから学習可能なノイズ定量化モデル Appropriateness Estimator を提案し，カリキュラムラーニングの実験を行った．実験においては，テストデータが人手で整備された Enron Subject データセットにおいてより効力を発揮することを明らかにした．本稿で提案した疑似負例により学習したモデルをカリキュラムラーニングに応用する手法は，翻訳タスクや対話タスクにも応用可能だと考えられる．

また，要約タスクにカリキュラムラーニングを応用する際，適切性と出力長の他に，抽出率や含意判定確率も有効であることを示した．今後の課題はこれらの指標を効果的に組み合わせ

る手法を開発することである．今回，ノイズの多い要約データセットに対するカリキュラムラーニングの有効性を調査したが，新聞記事などのノイズの比較的少ない要約データセットに対する調査は今後の課題となる．

5.3 本論文で取り組まなかった手法とその課題

本論文では教師データを使わない教師なし学習手法と，低品質教師データから学習する手法を取り上げた．他に高品質教師データが十分に得られない状況として，少数の高品質教師データのみがあるケースが考えられる．こうした状況に対処する手法として，少数の学習データから学習を行う Few-shot Learning がある．要約分野における Few-shot Learning として，大量に教師データがあるドメインで事前学習を行った後，少量の教師データしかないドメインで Fine-tuning を行う手法や (Fabbri et al. 2021)，や大量のデータで自己教師学習を行った後に，少量の教師データで Fine-tuning を行う手法 (Bražinskas et al. 2020a) などが提案されている．

前者の研究では，事前学習に使ったデータセットの内，Fine-tuning 用のデータに性質が近い（本文の参照要約に対する ROUGE-R など）データのサブセットを構築して Fine-tuning を行っており，後者の研究では，損失関数を用いてモデルの出力の性質が文体や長さ，レビューのレーティング等の観点から，Fine-tuning 用のデータに近づくように学習に制約をかけている．いずれの研究でも，何らかの観点で Fine-tuning 用のデータセットの性質を定義し，モデルがそうした性質に近い出力を行うよう制約をかけている．第 4 章では，要約の品質を表す指標を 4 つ取り上げ議論したが，Few-shot Learning でも同様に，Fine-tuning の際に近づける目的のデータセットの性質をどのように定量化すべきかが課題として残る．

低品質教師データから学習する手法として，本論文ではカリキュラムラーニングに取り組んだが，他にもフィルタリングを行う手法 (Qin et al. 2018; Csáky et al. 2019) や重要度に応じて学習データの損失関数に重み付けを行う手法 (Hu et al. 2019; Cai et al. 2020) が存在する．これらの手法もカリキュラムラーニング同様，フィルタリングの指標や重みを定量化することが必要であるため，要約の学習データの品質を反映した性質を適切に定量化する方法の模索が今後の課題として残る．また，これらの手法の内どの手法が最適であり，データセットによってどのように挙動が異なるかは明らかにされておらず，今後の研究による分析が必要となる．

5.4 今後の展望

本論文のテーマは高品質教師データが無い状況下における要約手法の研究であった．これに対する対処方法として，本論文第 3 章で提案した暗黙的引用を利用した要約モデル，第 4 章で取り上げたカリキュラムラーニング，また 5.3 節で取り上げた Few-shot Learning，フィルタリングや損失関数への重み付け手法，これらは全て要約の性質や要約に必要な要素を様々な側面から分析することで生まれた研究である．本テーマに関する研究の発展のためには，こうした性質の定性的な分析と，適切な定量化方法，及びモデルの出力を定量化した性質に近づける手法の考案が重要である．近年，新聞記事のみならず，メールやソーシャルメディア，チャットや科学論文，特許文書など様々なドメインに要約タスクの適用が広がっている．こうした多様なデータセットに対する分析から要約に求められる性質，データセットごとの性質の違いがより深く分析され，本テーマの研究が深まっていくことが期待される．

参考文献

Bahdanau, D., Cho, K., and Bengio, Y. (2015). "Neural Machine Translation by Jointly Learning to Align and Translate." In Bengio, Y. and LeCun, Y. (Eds.), *3rd International Conference on Learning Representations, (ICLR 2015)*.

Baziotis, C., Androutsopoulos, I., Konstas, I., and Potamianos, A. (2019). "SEQˆ3: Differentiable Sequence-to-Sequence-to-Sequence Autoencoder for Unsupervised Abstractive Sentence Compression." In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2019)*, pp. 673–681.

Bengio, Y., Louradour, J., Collobert, R., and Weston, J. (2009a). "Curriculum Learning." In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML 2009)*, pp. 41–48.

Bengio, Y., Louradour, J., Collobert, R., and Weston, J. (2009b). "Curriculum Learning." In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML 2009)*, pp. 41–48.

Bhatia, S., Biyani, P., and Mitra, P. (2014). "Summarizing Online Forum Discussions – Can Dialog Acts of Individual Messages Help?" In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pp. 2127–2131.

Bražinskas, A., Lapata, M., and Titov, I. (2020a). "Few-Shot Learning for Opinion Summarization." In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020)*, pp. 4119–4135.

Bražinskas, A., Lapata, M., and Titov, I. (2020b). "Unsupervised Opinion Summarization as Copycat-Review Generation." In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pp. 5151–5169.

Brin, S. and Page, L. (1998). "The Anatomy of a Large-scale Hypertextual Web Search Engine." *Computer Networks, vol. 30*, pp. 107–117.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse,

C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). "Language Models are Few-Shot Learners." *arXiv preprint arXiv:2005.14165*.

Cai, H., Chen, H., Song, Y., Zhang, C., Zhao, X., and Yin, D. (2020). "Data Manipulation: Towards Effective Instance Learning for Neural Dialogue Generation via Learning to Augment and Reweight." In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pp. 6334–6343.

Carenini, G., Ng, R. T., and Zhou, X. (2007). "Summarizing Email Conversations with Clue Words." In *Proceedings of the 16th International Conference on World Wide Web (WWW 2007)*, pp. 91–100.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation." In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pp. 1724–1734.

Chu, E. and Liu, P. J. (2019). "MeanSum: A Neural Model for Unsupervised Multi-Document Abstractive Summarization." In *Proceedings of the 36th International Conference on Machine Learning, (ICML 2019)*, pp. 1223–1232.

Cirik, V., Hovy, E., and Morency, L.-P. (2016). "Visualizing and Understanding Curriculum Learning for Long Short-Term Memory Networks." *arXiv preprint arXiv:1611.06204*.

Cohan, A., Dernoncourt, F., Kim, D. S., Bui, T., Kim, S., Chang, W., and Goharian, N. (2018). "A Discourse-Aware Attention Model for Abstractive Summarization of Long Documents." In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2018)*, pp. 615–621.

Csáky, R., Purgai, P., and Recski, G. (2019). "Improving Neural Conversational Models with Entropy-Based Data Filtering." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pp. 5650–5669.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). "ImageNet: A large-scale hierarchical image database." In *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pp. 248–255.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). "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 (NAACL 2019)*, pp. 4171–4186.

Edmundson, H. P. (1969). "New Methods in Automatic Extracting." *J. ACM*, **16** (2), pp. 264–285.

Erkan, G. and Radev, D. R. (2004). "LexRank: Graph-based Lexical Centrality As Salience in Text Summarization." *J. Artif. Int. Res.*, **22** (1), pp. 457–479.

Esplà, M., Forcada, M., Ramírez-Sánchez, G., and Hoang, H. (2019). "ParaCrawl: Web-scale parallel corpora for the languages of the EU." In *Proceedings of Machine Translation Summit XVII: Translator, Project and User Tracks*, pp. 118–119. European Association for Machine Translation.

Fabbri, A., Han, S., Li, H., Li, H., Ghazvininejad, M., Joty, S., Radev, D., and Mehdad, Y. (2021). "Improving Zero and Few-Shot Abstractive Summarization with Intermediate Fine-tuning and Data Augmentation." In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2021)*, pp. 704–717.

Feng, L., Shu, S., Lin, Z., Lv, F., Li, L., and An, B. (2020). "Can Cross Entropy Loss Be Robust to Label Noise?" In Bessiere, C. (Ed.), *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, (IJCAI 2020)*, pp. 2206–2212.

Fevry, T. and Phang, J. (2018). "Unsupervised Sentence Compression using Denoising Auto-Encoders." In *Proceedings of the 22nd Conference on Computational Natural Language Learning (CoNLL 2018)*, pp. 413–422.

Filippova, K. (2010). "Multi-Sentence Compression: Finding Shortest Paths in Word Graphs." In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pp. 322–330.

Fum, D., Guida, G., and Tasso, C. (1986). "Tailoring Importance Evaluation to Reader's Goals: A Contribution to Descriptive Text Summarization." In *Coling 1986 Volume 1: The 11th International Conference on Computational Linguistics (COLING 1986)*.

Ganesan, K. (2015). "ROUGE 2.0: Updated and Improved Measures for Evaluation of Summarization Tasks.".

Gholipour Ghalandari, D. (2017). "Revisiting the Centroid-based Method: A Strong Baseline for Multi-Document Summarization." In *Proceedings of the Workshop on*

New Frontiers in Summarization, pp. 85–90.

Goldberger, J. and Ben-Reuven, E. (2017). "Training deep neural-networks using a noise adaptation layer." In *5th International Conference on Learning Representations, (ICLR 2017)*.

Gu, J., Lu, Z., Li, H., and Li, V. O. (2016). "Incorporating Copying Mechanism in Sequence-to-Sequence Learning." In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, pp. 1631–1640.

Haghighi, A. and Vanderwende, L. (2009). "Exploring Content Models for Multi-Document Summarization." In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2009)*, pp. 362–370.

Hashimoto, K., Xiong, C., Tsuruoka, Y., and Socher, R. (2017). "A Joint Many-Task Model: Growing a Neural Network for Multiple NLP Tasks." In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, pp. 1923–1933.

He, Z., Chen, C., Bu, J., Wang, C., Zhang, L., Cai, D., and He, X. (2012). "Document Summarization Based on Data Reconstruction." In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI 2012)*, pp. 620–626.

Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., and Blunsom, P. (2015). "Teaching Machines to Read and Comprehend." In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R. (Eds.), *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, pp. 1693–1701.

Hirao, T., Isozaki, H., Maeda, E., and Matsumoto, Y. (2002). "Extracting Important Sentences with Support Vector Machines." In *Proceedings of the 19th International Conference on Computational Linguistics (COLING 2002)*, COLING '02, pp. 1–7.

Hochreiter, S. and Schmidhuber, J. (1997). "Long Short-Term Memory." *Neural Computation*, **9** (8), pp. 1735–1780.

Hu, Z., Tan, B., Salakhutdinov, R. R., Mitchell, T. M., and Xing, E. P. (2019). "Learning Data Manipulation for Augmentation and Weighting." In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (Eds.), *Advances in Neural Information Processing Systems (NeurIPS 2019)*, Vol. 32.

Isonuma, M., Mori, J., and Sakata, I. (2019). "Unsupervised Neural Single-Document

Summarization of Reviews via Learning Latent Discourse Structure and its Ranking." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pp. 2142–2152.

Jang, E., Gu, S., and Poole, B. (2017). "Categorical Reparameterization with Gumbel-Softmax." In *5th International Conference on Learning Representations, (ICLR 2017)*. OpenReview.net.

Kågebäck, M., Mogren, O., Tahmasebi, N., and Dubhashi, D. (2014). "Extractive Summarization using Continuous Vector Space Models." In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)*, pp. 31–39.

Kim, B., Kim, H., and Kim, G. (2019). "Abstractive Summarization of Reddit Posts with Multi-level Memory Networks." In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2019)*, pp. 2519–2531.

Kingma, D. P. and Ba, J. (2015). "Adam: A Method for Stochastic Optimization." In *3rd International Conference on Learning Representations, (ICLR 2015)*.

Klimt, B. and Yang, Y. (2004). "The Enron Corpus: A New Dataset for Email Classification Research." In *Proceedings of the 15th European Conference on Machine Learning (ECML 2004)*, pp. 217–226.

Kocmi, T. and Bojar, O. (2017). "Curriculum Learning and Minibatch Bucketing in Neural Machine Translation." In *Proceedings of the International Conference Recent Advances in Natural Language Processing, (RANLP 2017)*, pp. 379–386.

Koehn, P. (2004). "Statistical Significance Tests for Machine Translation Evaluation." In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pp. 388–395.

Kornilova, A. and Eidelman, V. (2019). "BillSum: A Corpus for Automatic Summarization of US Legislation." In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pp. 48–56, Hong Kong, China.

Koupaee, M. and Wang, W. Y. (2018). "WikiHow: A Large Scale Text Summarization Dataset." *arXiv preprint arXiv:1810.09305*.

Kryscinski, W., Keskar, N. S., McCann, B., Xiong, C., and Socher, R. (2019). "Neural Text Summarization: A Critical Evaluation." In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint*

Conference on Natural Language Processing (EMNLP-IJCNLP 2019), pp. 540–551.

Kumar, G., Foster, G., Cherry, C., and Krikun, M. (2019). "Reinforcement Learning based Curriculum Optimization for Neural Machine Translation." In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2019)*, pp. 2054–2061.

Kupiec, J., Pedersen, J., and Chen, F. (1995). "A Trainable Document Summarizer." In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, (SIGIR 1995), pp. 68–73.

Laban, P., Hsi, A., Canny, J., and Hearst, M. A. (2020). "The Summary Loop: Learning to Write Abstractive Summaries Without Examples." In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pp. 5135–5150.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). "Deep learning." *nature*, **521** (7553), p. 436.

Lee, J. and Chung, S. (2020). "Robust Training with Ensemble Consensus." In *8th International Conference on Learning Representations, (ICLR 2020)*.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2020). "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension." In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pp. 7871–7880.

Li, J., Li, H., and Zong, C. (2019). "Towards Personalized Review Summarization via User-Aware Sequence Network." In *The Thirty-Third AAAI Conference on Artificial Intelligence, (AAAI 2019)*, pp. 6690–6697.

Li, M., Soltanolkotabi, M., and Oymak, S. (2020). "Gradient Descent with Early Stopping is Provably Robust to Label Noise for Overparameterized Neural Networks.". Vol. 108 of *Proceedings of Machine Learning Research*, pp. 4313–4324.

Li, W., Wang, L., Li, W., Agustsson, E., and Gool, L. V. (2017). "WebVision Database: Visual Learning and Understanding from Web Data." *arXiv preprint arXiv:1708.02862*.

Lin, C.-Y. (2004). "ROUGE: A Package for Automatic Evaluation of Summaries." In *Text Summarization Branches Out*, pp. 74–81.

Lin, T., Goyal, P., Girshick, R. B., He, K., and Dollár, P. (2020). "Focal Loss for Dense

Object Detection." *IEEE Trans. Pattern Anal. Mach. Intell.*, **42** (2), pp. 318–327.

Liu, H., Yu, H., and Deng, Z.-H. (2015). "Multi-document Summarization Based on Two-level Sparse Representation Model." In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI 2015)*, pp. 196–202.

Liu, Y. and Lapata, M. (2019). "Text Summarization with Pretrained Encoders." In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3730–3740.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). "RoBERTa: A Robustly Optimized BERT Pretraining Approach." *arXiv preprint arXiv:1907.11692*.

Loshchilov, I. and Hutter, F. (2019). "Decoupled Weight Decay Regularization." In *7th International Conference on Learning Representations (ICLR 2019)*.

Loza, V., Lahiri, S., Mihalcea, R., and Lai, P.-H. (2014). "Building a Dataset for Summarization and Keyword Extraction from Emails." In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pp. 2441–2446. European Languages Resources Association (ELRA).

Luhn, H. P. (1958). "The Automatic Creation of Literature Abstracts." *IBM J. Res. Dev.*, **2** (2), pp. 159–165.

Ma, S., Deng, Z.-H., and Yang, Y. (2016). "An Unsupervised Multi-Document Summarization Framework Based on Neural Document Model." In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers (COLING 2016)*, pp. 1514–1523.

Ma, X., Huang, H., Wang, Y., Romano, S., Erfani, S., and Bailey, J. (2020). "Normalized Loss Functions for Deep Learning with Noisy Labels." In *Proceedings of the 37th International Conference on Machine Learning, (ICML 2020)*.

Matsumaru, K., Takase, S., and Okazaki, N. (2020). "Improving Truthfulness of Headline Generation." In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pp. 1335–1346.

Mehdad, Y., Carenini, G., and Ng, R. T. (2014). "Abstractive Summarization of Spoken and Written Conversations Based on Phrasal Queries." In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*,

pp. 1220–1230.

Meng, Y., Zhang, Y., Huang, J., Wang, X., Zhang, Y., Ji, H., and Han, J. (2021). "Distantly-Supervised Named Entity Recognition with Noise-Robust Learning and Language Model Augmented Self-Training." In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021)*, pp. 10367–10378.

Mihalcea, R. and Tarau, P. (2004). "TextRank: Bringing Order into Text." In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pp. 404–411.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). "Efficient Estimation of Word Representations in Vector Space." In *1st International Conference on Learning Representations, (ICLR 2013)*.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). "Distant supervision for relation extraction without labeled data." In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP (ACL 2009)*, pp. 1003–1011.

Nallapati, R., Zhou, B., dos Santos, C., Gu ☒ lçehre, Ç., and Xiang, B. (2016). "Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond." In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pp. 280–290.

Nguyen, D. T., Mummadi, C. K., Ngo, T., Nguyen, T. H. P., Beggel, L., and Brox, T. (2020). "SELF: Learning to Filter Noisy Labels with Self-Ensembling." In *8th International Conference on Learning Representations, (ICLR 2020)*.

Northcutt, C. G., Athalye, A., and Mueller, J. (2021). "Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks." *arXiv preprint arXiv:2103.14749*.

Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., and Auli, M. (2019). "fairseq: A Fast, Extensible Toolkit for Sequence Modeling." In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations) (NAACL 2019)*, pp. 48–53.

Oya, T. and Carenini, G. (2014). "Extractive Summarization and Dialogue Act Modeling on Email Threads: An Integrated Probabilistic Approach." In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL 2014)*, pp. 133–140.

Parikh, A., Täckström, O., Das, D., and Uszkoreit, J. (2016). "A Decomposable Attention Model for Natural Language Inference." In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pp. 2249–2255.

Platanios, E. A., Stretcu, O., Neubig, G., Poczos, B., and Mitchell, T. (2019). "Competence-based Curriculum Learning for Neural Machine Translation." In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2019)*, pp. 1162–1172.

Qin, P., Xu, W., and Wang, W. Y. (2018). "Robust Distant Supervision Relation Extraction via Deep Reinforcement Learning." In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, pp. 2137–2147.

Radev, D. R., Jing, H., Styś, M., and Tam, D. (2004). "Centroid-based summarization of multiple documents." *Information Processing Management*, **40** (6), pp. 919–938.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2018). "Language Models are Unsupervised Multitask Learners.".

Rossiello, G., Basile, P., and Semeraro, G. (2017). "Centroid-based Text Summarization through Compositionality of Word Embeddings." In *Proceedings of the MultiLing 2017 Workshop on Summarization and Summary Evaluation Across Source Types and Genres*, pp. 12–21.

Rush, A. M., Chopra, S., and Weston, J. (2015). "A Neural Attention Model for Abstractive Sentence Summarization." In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pp. 379–389.

Sandhaus, E. (2008). "The new york times annotated corpus." In *Linguistic Data Consortium 2008*.

See, A., Liu, P. J., and Manning, C. D. (2017). "Get To The Point: Summarization with Pointer-Generator Networks." In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pp. 1073–1083.

Sennrich, R., Haddow, B., and Birch, A. (2016). "Neural Machine Translation of Rare Words with Subword Units." In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, pp. 1715–1725.

Shang, G., Ding, W., Zhang, Z., Tixier, A., Meladianos, P., Vazirgiannis, M., and Lorré, J.-P. (2018). "Unsupervised Abstractive Meeting Summarization with Multi-

Sentence Compression and Budgeted Submodular Maximization." In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, pp. 664–674.

Sharma, E., Li, C., and Wang, L. (2019). "BIGPATENT: A Large-Scale Dataset for Abstractive and Coherent Summarization." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pp. 2204–2213.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). "Sequence to Sequence Learning with Neural Networks." In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K. Q. (Eds.), *Advances in Neural Information Processing Systems (NIPS 2014)*, Vol. 27.

Tarvainen, A. and Valpola, H. (2017). "Mean Teachers Are Better Role Models: Weight-Averaged Consistency Targets Improve Semi-Supervised Deep Learning Results." In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS 2017)*, pp. 1195–1204.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). "Attention is All you Need." In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (Eds.), *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pp. 5998–6008.

Wang, W., Caswell, I., and Chelba, C. (2019). "Dynamically Composing Domain-Data Selection with Clean-Data Selection by "Co-Curricular Learning" for Neural Machine Translation." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pp. 1282–1292.

Wang, W., Watanabe, T., Hughes, M., Nakagawa, T., and Chelba, C. (2018). "Denoising Neural Machine Translation Training with Trusted Data and Online Data Selection." In *Proceedings of the Third Conference on Machine Translation: Research Papers (WMT 18)*, pp. 133–143.

Wang, Y., Ma, X., Chen, Z., Luo, Y., Yi, J., and Bailey, J. (2019). "Symmetric Cross Entropy for Robust Learning With Noisy Labels." In *2019 IEEE/CVF International Conference on Computer Vision, (ICCV 2019)*, pp. 322–330.

Williams, A., Nangia, N., and Bowman, S. (2018). "A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference." In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics:*

Human Language Technologies (NAACL 2018), pp. 1112–1122.

Yin, W. and Pei, Y. (2015). "Optimizing Sentence Modeling and Selection for Document Summarization." In *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI 2015)*, pp. 1383–1389.

Zhang, B., Nagesh, A., and Knight, K. (2020). "Parallel Corpus Filtering via Pre-trained Language Models." In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pp. 8545–8554.

Zhang, R. and Tetreault, J. (2019). "This Email Could Save Your Life: Introducing the Task of Email Subject Line Generation." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pp. 446–456.

Zhang, Z. and Sabuncu, M. R. (2018). "Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels." In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (Eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, (NeurIPS 2018)*, pp. 8792–8802.

Zhao, L., Xu, W., and Guo, J. (2020). "Improving Abstractive Dialogue Summarization with Graph Structures and Topic Words." In *Proceedings of the 28th International Conference on Computational Linguistics (COLING 2020)*, pp. 437–449.

Zheng, H. and Lapata, M. (2019). "Sentence Centrality Revisited for Unsupervised Summarization." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pp. 6236–6247.

Zhong, M., Wang, D., Liu, P., Qiu, X., and Huang, X. (2019). "A Closer Look at Data Bias in Neural Extractive Summarization Models." In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pp. 80–89.

Zhou, Y., Yang, B., Wong, D. F., Wan, Y., and Chao, L. S. (2020). "Uncertainty-Aware Curriculum Learning for Neural Machine Translation." In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pp. 6934–6944.

博士課程での研究業績

論文誌

- 狩野竜示, 谷口友紀, 大熊智子 (2022) “要約データの適切性定量化を利用したカリキュラムラーニング”, 自然言語処理, 29(1), (予定).
- 狩野竜示, 谷口友紀, 大熊智子 (2021) “会話文における暗黙的引用を利用した抽出型教師なし要約”, 自然言語処理, 28(2), p532-553.

国際会議論文

- Ryuji Kano, Takumi Takahashi, Toru Nishino, Motoki Taniguchi, Tomoki Taniguchi, Tomoko Ohkuma (2021) “Quantifying Appropriateness of Summarization Data for Curriculum Learning”, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics* (EACL 2021), p1395-1405.
- Ryuji Kano, Yasuhide Miura, Tomoki Taniguchi, Tomoko Ohkuma (2020) “Identifying Implicit Quotes for Unsupervised Extractive Summarization of Conversations”, *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing* (AACL-IJCNLP 2020), p291-302.

その他の研究業績

国際会議論文

- Ryuji Kano, Yasuhide Miura, Motoki Taniguchi, Yan-Ying Chen, Francine Chen, Tomoko Ohkuma (2018) “Harnessing Popularity in Social Media for Extractive Summarization of Online Conversations”, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (EMNLP 2018), p1139-1145.

国内会議論文

- 狩野竜示, 谷口友紀, 大熊智子 (2020) “フィルタリングによるタイトル-本文ペアの要約教師データの改善”, 言語処理学会第 26 回年次大会 (NLP2020).
- 狩野竜示, 三浦康秀, 大熊智子 (2018) “オンライン掲示板の反応予測に有効なテキスト素性の分析”, 言語処理学会第 24 回年次大会 (NLP2018).
- 狩野竜示, 谷口元樹, 根本啓一, 大西健司, 大熊智子 (2017) “文脈情報を考慮した旅行中ツイートの判別”, 言語処理学会第 23 回年次大会 (NLP2017).

謝辞

本論文は，筆者である狩野竜示が東京工業大学大学院工学院情報通信コースに在籍中に行った研究の成果をまとめ上げたものです．本論文の審査を引き受けてくださった篠崎隆宏准教授，熊澤逸夫教授，中山実教授に感謝申し上げます．

指導教員である奥村学教授には研究の方向性についての多くの助言をいただきました．国際会議の論文としてまとめ上げるのに必要な要素は何か，研究のテーマが一貫しているかをはじめとして多くのアドバイスをいただきました．また，投稿論文が不採択になった時には励ましの言葉をいただき大いに奮起することができました．ここに感謝の意を表します．産業技術総合研究所に務められている高村大也教授からは，国際会議の論文としてまとめ上げるのに必要な貢献や，実験結果の分析結果の解釈をはじめとして多くのご助言をいただきました．奥村・船越研究室の船越孝太郎准教授には論文や発表資料に関して様々なアドバイスをいただきました．要点をついた鋭いご指摘をいただき，短期間で論文や発表資料の完成度を上げることができました．上垣外英剛助教授からは毎月のゼミにおいて，研究の方向性についてアドバイスをいただきました．上垣外助教授は自然言語処理のあらゆる分野に精通しておられ，様々な知識を活用して多方面からアドバイスをくださり，私自身も自分の専門範囲以外の勉強も広く行わなければならないということを実感させられました．

富士ゼロックス株式会社（現・富士フイルムビジネスイノベーション株式会社）の上司，先輩，同僚の方々からは様々なご指導や刺激をいただきました．根本啓一さんには，自然言語処理のアルゴリズムやそれを用いたサービスの基本的な考えを教わりました．機械学習の門外漢であった私が自然言語処理の分野に足を踏み入れるきっかけをくださいました．大熊智子さん，谷口友紀さんには研究の進め方や論文の書き方について幅広くご指導いただきました．自然言語処理分野の論文を執筆したことのなかった私が博士号を取得するに至れたのはお二方の辛抱強いご指導によるものだと痛感しております．同じ研究室の先輩でもある三浦康秀さんには研究上のアドバイスに加えて，大学院の生活に関しても助言をいただきました．次々と大きな成果を発表する三浦さんの研究姿勢は，私の向上心を大いに奮い立たせてくれました．

最後に，生活面，精神面において広く支えとなってくれた両親に感謝いたします．両親の支

えなしには，博士号どころか学士号を取得することも不可能でしたし，高校受験や大学受験も乗り切ることはできませんでした．長きに渡る支えのおかげで，今回その成果を博士号として結実させることができました．ありがとうございました．