発話内容を用いたクエリ拡張による過去会話ログの検索

田貝　奈央†　　加藤　　誠††

† 筑波大学 知識情報・図書館学類　〒 305–8550 茨城県つくば市春日 1–2
†† 筑波大学 図書館情報メディア系　〒 305–8550 茨城県つくば市春日 1-2
E-mail: †s2113509@s.tsukuba.ac.jp, ††mpkato@acm.org

あらまし　本論文では，会話を円滑に進めるために必要な情報を，現在の会話内容に基づいて，会話ログから検索する手法を提案する．本研究における課題は，過去ログを検索する際に現在の発話内容だけをクエリとして用いた場合，次に行う会話に合致した内容が得られないという点である．提案手法では，事前学習言語モデル GPT-2 を用いて，現時点までに得られた会話内容から次に出現する単語を予測することでクエリ拡張を行った．これによって，文脈から次の会話内容を予測し，現時点でなく，次時点において適合する会話ログを得ることを期待した．実験では，YouTube で公開されている複数回にわたる会話を行っている動画を元に評価用データセットを作成し，提案手法の評価を行った．

キーワード　音声クエリ，ユーザ支援，質問応答，デジタルツイン

1　はじめに

一度聞いた情報を，別の機会に言及することは，特別感や信頼感を与える効果的なコミュニケーション手法として知られている [22]．例えば，会話の中でスポーツの話題が上がった際に，過去の会話を覚えておけば，会話相手の贔屓にしているチームなどの情報を思い出しつつ会話に組み込むことができる．しかし，会話を覚えておくことは人によって得意・不得意がある．また，得意な人でも，コミュニケーション量が多くなると覚えておくことが難しくなる．この課題を解決するために，過去の会話で言及した現在の会話内容に関連する情報を会話中に閲覧できるシステムを実現する．これにより，特に多くの人と会う機会がある人や相手との信頼関係が重要となる職種に対してコミュニケーションの支援を行うことを目的とする．

会話を円滑に進めるために，過去の会話で言及した現在の会話内容に関連する情報を会話中に抽出する技術は，既存の音声ドキュメント検索の一般的な設計に基づき (1) 会話の常時記録，(2) 会話の自動書き起こし，(3) 現在の会話に基づく過去の会話ログの検索，(4) 過去の会話ログの要約，から構成される．これにより，会話中に現在話している内容を入力とし，過去の会話ログを抽出し提示することでコミュニケーションの補助を実現することができる．本研究で注目する「(3) 現在の会話に基づく過去の会話ログの検索」は，現在の発話内容と過去の会話ログが与えられたときに，現在の発話内容と関連する会話ログ中の発話を取得することを目的に行われる．(1),(2),(4) は，既存の技術で達成できるが，(3) には次のような課題が想定される．まず，発話終了後に関連する会話ログを利用者に発話を提示しても利用者が内容を把握するまでに時間を要する．これはユーザが提示情報を取捨選択し，次に行う会話を構成する必要があるためである．また，会話は流動的なもので，発話終了まで待つとトピックが変化し，提示情報を活かした会話ができないことも考えられる．しかし，現在の発話内容だけをクエリとして用い過去会話ログの検索を行った場合，次に行う会話に合致した内容が得られないといった問題がある．そのため，できる限り少ない現在の発話内容に基づいて，次に話題になるであろう内容を予測し，その話題に関連する会話を検索できることが望ましい．

本論文では，現在までの発話内容の文脈に基づいて，その後の発話文を予測することで，過去の会話から必要な情報を先回りして抽出する手法を提案する．これにより，発話文の少し後の文脈を予測し，過去の会話ログを検索することができるため，会話中のリアルタイムな検索を実現することが可能になる．

我々は，**GPT-2 を用いた発話文によるクエリ拡張**を用いて本提案を実現する．前の文脈を入力すると，以降に起こりやすいような単語列を予測可能なモデルである GPT-2 を利用し，将来の話題を予測して，時系列的に後に必要となるであろう会話を会話ログから抽出することで実現する．既存研究では，一般的な文書検索において GPT-2 を用いたクエリ拡張は効果があるとされている．しかし，発話文自体を入力とするのではなく，ユーザが入力したクエリに基づいて文書を生成し，クエリ拡張に用いるものである．本研究では，発話途中の文書を入力にし，その次に発話されると予測される文書を生成しクエリ拡張に用いることで検索精度が向上すると考える．

提案手法の有意性を検証するためには，複数回に渡り，過去の会話に言及しているデータセットが必要だが，それに適したデータセットが入手できなかったため，新たにデータセットを作成した．データセットの作成には，YouTube 上で公開されている二人での会話を複数回に渡り行なっている動画を活用した．本動画は，過去の回に言及するシーンが複数回存在するため，本研究のタスクを評価するのに相応しいと考えられる．データセットの作成は次の手順でおこなった．まず，自動でテキストデータに書き起こし，動画内で複数回出現する固有名詞で検索を行い，単語が出現するパッセージのリストを作成した．その発話文の集合に対し，スコア付けを行い，ある時点の発話文に関連する過去の発話文の紐付けを行い，発話途中文書であるク

エリと適合発話文書集合のペアを作成した．本データセットの統計情報を示し，提案手法の評価に活用する．実験では，このデータセットを用い，ベースライン手法と提案手法を比較し，提案手法の有効性を検証した．また，検索に用いるパラメータの変化を行うことで実験結果の評価をおこなった．実験を行った結果，提案したクエリ拡張手法により検索結果が統計的有意に改善することを示した．また，生成文書数や文脈として利用する文脈窓の大きさなどのパラメータの変化によって精度が変化することを示した．

本論文の貢献は以下のとおりである．

（1） 発話文を用いた過去の会話を抽出する技術を実現するための問題設定を提案した．

（2） 将来の話題を予測し，時系列的に後に必要となるであろう会話を会話ログから検索する手法を提案した．これは，発話途中の文脈を利用し，以降に起こりやすいような単語を予測できるモデルである GPT-2 を用いて実現する．

（3） YouTube 上で公開されている動画を用い，複数回に渡り会話をおこなっているデータセットを作成し，提案モデルの評価を行った．これにより，発話文を用いて過去の会話を検索するタスクにおいて，提案モデルの有効性を示した．

本論文の構成は以下の通りである．2 節では過去の会話検索，発話を用いた対話型検索システム，会議ブラウジング，および，クエリ拡張の関連研究について述べる．3 節では問題設定を説明し，発話文予測を用いたクエリ拡張手法，および，その主問題への適用方法について述べる．4 節では実験結果を示す．最後に，5 節では今後の課題と共に本論文の結論を述べる．

2 関 連 研 究

本節では，過去の会話検索，発話を用いた対話型検索システム，会議ブラウジング，および，クエリ拡張の関連研究について述べる．

2.1 過去の会話検索

１つ目は，過去の会話検索である．人々は日々膨大な量のコミュニケーションを行なっており，重要な決定や判断などを行なっている．日常生活で起こったことを覚えておくことは，日常生活を円滑に送る上で重要になるが，出来事や情報を記憶することに対する得手不得手には個人差がある．そのため，人々はできるだけ多く重要な出来事や情報を記録し，再び閲覧できるような工夫を重ねている．しかし，現在提供されているテクノロジーで行える人間の記憶領域の支援はリマインダーやカレンダーの予定程度しかない．また，記録は人々が手動で行うことが多く，記録忘れが発生しやすい．

そこで，注目されるのが現在の技術によって，日常生活の体験を長時間自動的にキャプチャしたデータであるライフログである．しかし，保存したライフログは膨大な量になるため，一つの情報を得るために全てのライフログを閲覧することは難しい．また，提示情報が多すぎると混乱させる可能性があるため．ユーザが思い出したい記憶を想起させるきっかけとなる情報を提示する必要がある．そこで，Yi と Gareth は過去の会合の要約情報や，センサ情報から環境情報に関するタグを提示することでライフログへ手軽にアクセスできるシステムを提案した [4].

また，Seyed と Fabio は，動画やセンサデータから，人々が忘れやすそうな部分を推測して提示するパーソナルアシスタントを提案した [1, 2]．記憶に関する病気のない健康的な人々でも，一日に接する情報の数が多かったり，注意力が足りなかったりすることで，過去の重要な出来事を思い出せないことがある．そこで，一定期間内の会話のうちどの部分を忘れそうか推測し，それを支援するリマインダー機能のようなシステムを構築する．記憶に残る，残らない部分を予測するモデルを提示した．

これらの研究は，ライフログという膨大なデータを用いて人間の記憶の拡張を行えるが，　必要な情報を取り出す際に，自ら検索クエリを入力したり，思い出したい記録を選ぶ必要があった．そのため，会話中に過去の情報にアクセスすることは難しい．そこで，本研究では，音声データで記録された過去の会話を発話文を用いて検索することに注目する．

Kunpeng らは，チャットツールの過去の会話に対し検索を行う「会話を検索する」という新しいタスクの提案と，その実験を容易にする研究システムの開発をおこなった [15]．与えられた会話から類似する過去の会話を検索するタスクは，会話のトピックが以前に議論されていることを示すことで，時間の浪費を防いだり，重要な決定を思い出したり追跡するのに役立つ．

これらの研究では，チャットツールでやりとりされた過去の会話を検索するタスクに取り組んでいる．本研究では，口頭での会話を対象にしているが，現在の会話に関連する過去の会話を抽出することで得られるメリットはテキストか口頭であるかには依存しないと考える．また，発話をテキスト化することで，既存研究の手法を適用することもできると考える．しかし，テキストでの会話と口頭での会話の違いは，発話文をテキストに変換する際の発話文書き起こしツールの精度により，意図しない単語が入力されることもあり，ノイズがあるテキストに対する検索になることを考慮に入れる必要がある．

TREC の SDR (Spoken Document Retrieval) トラックでは，音声録音のアーカイブに対しアドホック検索を実現するタスクが提案された [8]．一般的な SDR は，自動音声認識と情報検索技術の組み合わせにより実現する．まず，音声ドキュメントを音声認識エンジンを利用し，タイムスタンプ付きの書き起こし文書に変換する．次に，書き起こし文書は検索システムによって索引付けされ，検索できる．クエリに対し返却される結果は，関連する書き起こし文を指すタイムスタンプのリストになっており，指している音声の内容とクエリ間の類似性スコアでランク付けされる．

発話意図を自ら話すものや，検索クエリをユーザが直接入力するものはあるが，発話中に過去の発話文を検索するタスクは存在しない．

2.2 発話を用いた対話型検索システム

発話を用いた検索システムの構築はさまざまな研究で取り組

まれている [12]．発話文からユーザの発話意図を読み取る研究は，保険代理店や医療，ヘルプデスクなどさまざまな場面でのニーズに応じて研究が進められている [7, 9, 11, 13, 18]．

Sosuke らは，会話文の分析を行い，共同検索の会話で表現される情報ニーズをモデル化する方法を提案し，会話中の情報検索の必要性を示した [17]．本研究では，発話の 17%が共同検索タスクで意識的なニーズまたは形式化されたニーズを表現することがわかった．残りの 80%を活用することで，従来のクエリ動作では表現されないユーザの情報ニーズを掘り起こす有望な情報源になると言われている．この研究では，共同検索タスクでの会話のニーズを分析しているが，本研究で目的としている，自然会話中での検索で示される意識的または形式化されたニーズはもっと少なくなる．

Nan らは，現在の会話に関連する文書をプロジェクターで机に投影するシステムを提案した [10]．この研究は，従来のタスク思考の検索と異なり，会議中にユーザが検索を行うこと（日和見検索）をサポートするために発話文から検索クエリの候補を提示し，ユーザに選択させる．

Andre らは，会話を監視し，自動認識された単語を用いてユーザが現在利用しやすいマルチメディアの提示を行う自動コンテンツ連携装置を提案した [14]．

これらの研究は，会話中に検索を行えるといった類似点があるが，会話中に自ら検索クエリを選択する必要があること，検索先が web 上の情報であることが異なる点である．実際の想定される利用シーンは個人間での会話であるため，コミュニケーション自体を中断させないことや web 上に存在しない重要な情報に対してアクセスできることが望ましい．そこで，我々の提案システムは，会話の流れに応じて逐次的に過去の会話を提示する必要がある．

2.3 会議ブラウジング

コンピュータの性能向上や保存容量の増大により，会議の議事録をデータを記録することが増えてきた．すると，蓄積されたデータから必要な情報を効率よく取り出す手法の考案が必要になった．これは，音声・映像データを見返すには，時間がかかってしまうといった問題点があるからだ．そこで，文字起こしを行ったデータをデータとして蓄積し，文書検索のように，クエリを入力して必要な情報を取り出す手法が提案されるようになった．

音声分割、索引付け、および音声認識を利用した会議の検索と閲覧の方法はいくつか存在する [3]．基本的に音声認識技術を用い，音声データをテキストデータに変換し，既存の文書検索手法が利用できるようになっている．

Alex らは人間の会議の記録を書き留め，迅速にアクセスするためのシステムである Meeting Browser を提案した [20]．このシステムは，(1) 音声文字起こしエンジン (2) 特徴的な会話を抽出する要約ツール (3) 発話行為の判定 (4) 発話者の特定の 4 つのコンポーネントで構成されている．この研究では，記録された会議を検索する際に，自動書き起こしを行った文書から自動で生成される要約がトピックを認識することに有用であることが示された．

John らは大規模音声コーパスから音声文書を検索・閲覧するシステムである SCAN を提案した [5]．この研究の特徴は，クエリ拡張をおこなっている点である．ユーザクエリは短いことが多いため，関連するクエリを追加して検索を行うことで検索精度の向上を図っている．手法としては，最初に入力されたユーザクエリと関連度の高いドキュメントを特定し，これらのドキュメント内で頻繁に使用される単語をユーザクエリに追加することでクエリ拡張を実現する．TREC-7 SDR トラックデータを用いてこの手法を用いたクエリ拡張の効果を示している．

これらの研究と我々の研究は共に会話内容を音声認識などでテキストとして記録した過去の会話ログに対して検索を行い，必要な部分を抽出するといった点で類似するが，我々の提案手法は現在の発話文を用いて検索を行うものであり，ユーザがクエリを入力したり，会話文全てを入力するものではない．

また，本論文では現在発話している内容までの文脈を利用し，その後に出現しやすそうな単語を予測しクエリを作成することにより，発話文に関連する過去の会話のリアルタイムでの検索を実現する．円滑なコミュニケーションの補助には，リアルタイムな検索が不可欠だが，他の提案手法では，発話が終了したタイミングで検索を行うため，ユーザの情報欲求とずれが生まれてしまう．

2.4 クエリ拡張

一般的な情報検索ではユーザがクエリを入力することで自身の情報欲求を表現する．しかし，クエリの語彙が欲しい文書と異なる場合があり，クエリを文書と一致させることは難しい．また，クエリが短いと情報欲求を十分に満たしたクエリを表現することができず，検索システムの精度は低下する．クエリ拡張とは短いクエリをより大きなテキストに変換し，文書コレクションから関連文書を照合しやすくすることによってこれらの問題を改善する方法である．

Nikos らは，複数回の会話を用いた検索タスクにおいて，質問の文脈を用いたクエリ拡張を行う手法を提案した [19]．同じ単語でも，会話の文脈によって異なる意味で使われていることがあるので，一つのターンの発話からクエリを生成するのは不十分である．そのため，過去のターンの発話文に登場した単語に対し，検索クエリに加えるかどうかを二項分類問題として判断するモデルを構築する．

渡邉の研究では，会話，会議，見ている文章などをリアルタイムに文字起こしし，クエリを生成し自動的に検索結果を返す自動検索システムの提案を行った [21]．クエリを自動生成するために 3 レイヤー構造を導入し，検索ワードの前何ワードかを入力とすることで，文脈を考慮した文書検索を行った．この研究では，入力文書をそのまま利用するのではなく，重要な単語を抽出しクエリを生成することで検索精度の改善を行なっている．

Vincent の研究は，最近のテキスト生成モデルを使用して生成されたテキストをクエリ拡張に使用できることを示した [6]．

関連研究では，前のトークンから次のトークンを予測するような教師なし学習でトレーニングされたトランスフォーマーである GPT-2 [16] を使用し，元のクエリを生成モデルの入力とし，生成されたテキストをクエリ拡張に用いている．文書生成モデルを用いた文書の生成は，コストや時間がかかるため，できる限り少ない生成数で良い精度が達成したい．この研究では，クエリあたり 20 文書以上生成しても精度が変化せず，安定することを示した．また，重みづけによるクエリ拡張より良い精度を得られることを示している．

既存研究と本研究の違いについて述べる．既存研究では，与えられたユーザクエリごとに全く新しい文書を生成し，新しいクエリとして用いられる．本研究では，今まで発話された文書を入力として用い，まだ発話されていない部分を GPT-2 を用いて予測を行い，現在までの発話から出現しやすい単語を入力クエリとして用いることができる．複数ターンに渡る会話の文脈を入力できるため，生成される文書も発話の文脈を考慮した予測文書になると考えられる．

3 提 案 手 法

本節では，円滑なコミュニケーションの支援を行うシステムの概要と，現在の会話に関連する過去の会話を抽出する技術を実現する上での課題について説明を行う．それから，発話文予測を用いたクエリ拡張を導入し，その主問題への適用について述べる．

3.1 システム概要

図 1 に現在の会話に関連する過去の会話を提示することで円滑なコミュニケーションの支援を行うシステムの概要を示す．現在の会話に関連する過去の会話を抽出する技術は，(1) 会話の常時記録，(2) 会話の自動書き起こし，(3) 現在の会話に基づく過去の会話ログの検索，(4) 過去の会話ログの要約から実現する．

ユーザの会話は，マイクを用いてシステムに音声データを入力し，会話の常時記録を行う．この時，音声データをそのまま保存するとデータ量が膨大になってしまうため，音声認識ツールを用い，自動で発話文を書き起こし，単語分割することで過去の会話インデックスを作成する．このインデックスが，検索対象の文書集合となる．次に，現在のユーザの会話を録音し，逐次音声認識ツールを用い発話音声をテキストデータに変換する．このテキストデータをもとにクエリを作成し，現在の会話に関連する過去の会話文書のランキングを得る．全てをユーザに提示すると，情報量が多かったり，書き起こし時の誤りにより混乱してしまう可能性がある．そこで，得られた文書集合を要約し関連会話情報を作成し，ユーザに提示する．これによりリアルタイムの過去会話検索システムを実現する．(1),(2),(4) は，既存の技術で達成できるが，(3) には次のような課題が想定される．まず，発話終了後に関連する会話ログを利用者に発話を提示しても利用者が内容を把握するまでに時間を要する．これはユーザが提示情報を取捨選択し，次に行う会話を構成する必要があるためである．また，会話は流動的なもので，発話終了まで待つとトピックが変化し，提示情報を活かした会話ができないことも考えられる．しかし，現在の発話内容だけをクエリとして用い過去会話ログの検索を行った場合，次に行う会話に合致した内容が得られないといった問題がある．そこで，本研究では「(3) 現在の会話に基づく過去の会話ログの検索」に注目する．

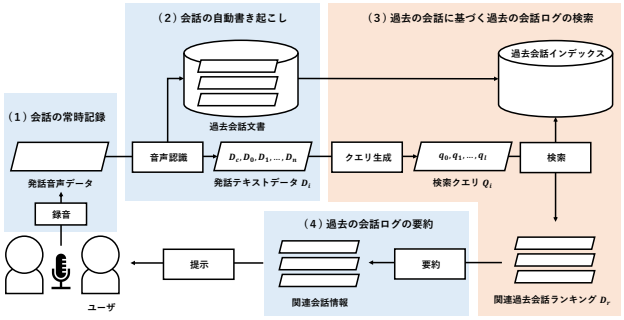

図 1　システム概要

3.2 問 題 設 定

本研究でのタスクは，現在の発話内容に関連する過去の会話内容を，発話途中の文章から検索することである．全ての発話文集合 $D = \{d_1, \ldots, d_s, \ldots, d_t, \ldots, d_T\}$ を定義する．ただし，それぞれの発話文 d_t は，ある時刻 $t = \{1, \ldots, s, \ldots, T\}$ に発話された発話文を表す．また，s は現在までの発話文の始まりを示す.「(3) 現在の会話に基づく過去の会話ログの検索」では，現在までの発話文集合 $D_c = \{d_s, \ldots, d_t\}$ を入力とすると，過去の発話文集合 $D_p = \{d_1, d_2, \ldots, d_{s-1}\}$ から関連文書集合 D_r を取り出すことを目的とする．

会話中に過去の会話ログを提示し，円滑な会話を実現するには，会話に応じて逐次的に検索をする必要がある．また，雑談を含む会話での発話文は短いものも多く，あるタイミングの発話文のみを入力とすると文脈が十分に考慮できない可能性がある．従って，現在の発話内容と関連する過去会話ログの検索に利用可能なのは，ある時点 t までに発話された文 $d_t \in D_c$ と，その時点を k 回遡った発話文を入力文書集合 $D_i = \{d_{t-k}, d_{t-(k-1)}, \ldots, d_t\}$ として利用する．ただし，$t \geq k$ である．

我々の主目的は，記録された過去の会話ログ D_p の中から，現在の会話 D_c に関連する過去の発話文書集合 $D_r (\subseteq D_p)$ を抽出することである．ただし，全ての発話文集合 D は,「(2) 会話の自動書き起こし」の際に発話終了時の無音状態を検出するまでを発話単位とする．我々は，過去に言及したことのある固有名詞が再び会話上に現れた際に，過去の発話文が抽出されることを期待する．これにより，固有名詞以前の発話内容から，適合する過去の会話ログが抽出できたか判定することで，検索モデルの評価を行う．

3.3 発話文予測を用いたクエリ拡張

本節では直近の発話の文脈から生成された予測発話文を活用したクエリ拡張手法を提案する．まず，GPT-2 を用いた予測発話文の生成とクエリ拡張方法を導入し，次に発話予測文を用いたクエリ拡張による関連会話文の検索方法について説明する．

リアルタイムに関連する過去の会話を検索するために，発話途中の文章の次に出現する単語を GPT-2 を用いて生成する．GPT-2 は，Transformer をベースに事前学習やファインチューニングをすることで非常に高い精度文章を生成するモデルである．

入力として書き起こされた発話途中の文書 D_t とその文脈にあたる k 回にわたる発話文の集合 $D_i = \{d_{t-k}, d_{t-(k-1)}, ..., d_t\}$ を GPT-2 に与え，文書の生成を行う．この時 $D_i \subseteq D$，$t \geqq k$ であることに注意する．入力発話集合をもとに GPT-2 モデルによって複数のテキストを生成する．この時の出力は指定された m 語からなる予測発話文 $\hat{d_j} = \{w_1, ..., w_m\}$ である．GPT-2 を用いたテキスト生成は決定論的ではないため，同じ入力発話文集合 D_i に対し，l 回の生成プロセスを実行する．これらの生成された複数の予測発話文の集合 $D_g = \{\hat{d_1}, \ldots, \hat{d_j}, \ldots, \hat{d_l}\}$ を用いてクエリの作成を行う．

発話文予測を用いたクエリ拡張方法を二つ提案する．1 つ目は，現在までの文脈から生成された予測発話文をもとにクエリを生成し，関連会話文の検索を行う．予測発話文は現在までの文脈が 2 つ目は，予測発話文と現在までの発話文からクエリを生成し，関連会話文の検索を行う．それぞれの方法のクエリ拡張方法を以下に示す．: (1) 入力に用いた発話文集合の D_i に対し分かち書きを行い，単語の抽出を行う．これにより，できるだけ現在の発話に近い文脈を検索クエリにすることができる．(2) 生成されたそれぞれの予測発話文 D_o に対し分かち書きを行い，単語の抽出を行う．これにより，生成された文書内でも，次に会話のトピックになりやすい特徴をクエリに組み込むことができる．(3) (1) で取得した単語の末尾 w 件と (2) で取得した予測発話文 D_o ごとの単語の先頭 w 件を結合し検索クエリとする．これにより，発話済みの文脈と，発話が予測される文脈のどちらも考慮したクエリを作成することができる．

これらの方法により，一つの入力発話文集合 D_i から予測された j 番目の予測発話文 $\hat{d_j}$ を元に名詞を抽出し，単語 $q_1, \ldots, q_i, \ldots, q_l$ を含む検索クエリ Q_j を生成する．これを全ての予測発話文集合 D_g に対し行うことで，入力クエリ集合 $\mathcal{Q} = \{Q_1, \ldots, Q_j, \ldots, Q_l\}$ を生成する．次に，発話予測文を用いたクエリ拡張による関連会話文の検索方法について説明する．前述の通り，生成された予測発話文それぞれから検索クエリを作成し，BM25 を用いて，過去の発話文集合 D_p との関連度を算出する．式 1 に BM25 の式を示す．文書 d は過去の会話文書集合 D_p の要素，クエリ Q は予測発話文 $\hat{d_g}$ から生成した単語 $q_1, \ldots, q_i, \ldots, q_n$ を含む検索クエリである．k_1 は 0 以上の実数，b は 0 から 1 の実数をとるパラメーターである．$\mathrm{idf}(q_i)$ は単語 q_i の idf，$f(q_i, d)$ は発話文章 d 中での単語 q_i の出現頻度，$|d|$ は発話文書 d の単語数，$\mathrm{avg}(dl)$ は全文章の平均単語数を示す．

$$\mathrm{score}(Q, d) = \sum_{\mathrm{i=1}}^{\mathrm{n}} \mathrm{idf}(q_i) \times \frac{(k_1 + 1) f(q_i, d)}{f(q_i, d) + k_1 (1 - b + b \frac{|d|}{\mathrm{avg}(dl)})} \tag{1}$$

次に，発話予測を用いたクエリ拡張時の文書の関連度スコアの式を式 2 に示す．入力クエリ集合 $\mathcal{Q} = \{Q_1, \ldots, Q_j, \ldots, Q_l\}$ と，過去の発話文 $d \in D_p$ を入力に与えると，d とそれぞれの検索クエリの間の BM25 スコアの合計が算出できる．これにより，複数のクエリによる検索結果の結合を行うことで，どのクエリでも抽出される過去発話文書のスコアを大きくする．

$$f(\mathcal{Q}, d) = \sum_{j=1}^{l} \mathrm{score}(Q_j, d) \tag{2}$$

関連度スコアの高い順上位 k 件を現在の会話に関連する過去の会話集合 D_r として提示する．現在の発話に関連する会話検索に発話文予測を用いたクエリ拡張手法により得られた検索結果となる．

4 実　　験

我々のタスクでは実際の利用状況に近い複数回に渡り，過去の会話に言及しているデータセットが必要になるが，適したデータセットが存在しないため，新たにデータセットを作成する．まずデータセットの作成の概略と統計情報について説明する．それからベースライン手法を含む実験設定について紹介し最後に実験結果を示す．

4.1 データセットの作成

我々は，円滑な会話を実現するために，現在の会話に関連する過去の会話を抽出するモデルを提案した．実際には，発話途中に検索を行うことで会話に応じたリアルタイムでの検索が求められる．そのため，途中の発話文を入力とした場合の関連発話文章を示す複数回にわたり過去の会話に言及しているデータセットを用い，提案モデルの評価を行う必要がある．

4.1.1 検索対象の文書コレクション

我々は，YouTube にて公開されている「ゆる言語学ラジオ」[1] というチャンネルの投稿動画を検索対象として設定した．本動画は,「言語学」に関する話題について二人で話すコンテンツとなっている．(1) 同一人物との会話が複数回に渡って記録されており，人物名や固有名詞の説明のような (2) 過去の回で言及した内容について再度言及することがあることから，我々のタスクにふさわしい動画であるといえる．

検索対象のコレクションの作成方法を説明する．まず，投稿順に 58 本（総時間：35 時間 55 分 18 秒）の動画を保存し，自動でテキストデータに書き起こす．書き起こしには Azure の Speech to Text を用いた．一般的な雑談に対する書き起こしの精度は 90%である．次に，書き起こしたテキストを発話文単位で分割し，発話文書 id を割り当て記録する．本実験での発話

1：https://www.youtube.com/@yurugengo

表 1　スコアの条件と例

	条件	例
-1	完全に異なる文書	自動書き起こしでの誤り
0	関連度が低い文書	部分的に関連する文書
1	関連する文書	完全に関連する文書
2	入力に用いる文書	入力文に用いる文書（特定の単語が含まれる発話文の中で最後に発話されたもの）

表 2　自作データセットの統計情報

	自作データセット
全体の発話文書数 D	6332
平均発話文書長 D	136.03
入力発話途中文書数 d_t	39
平均入力発話途中文書長 d_t	101.49
入力発話途中文書 (d_t) ごとの平均適合発話文書数 D_r	12.68

単位は，Azure の Speech to Text の発話定義に基づき次のように定義する．:(1) 発話終了時の無音状態を検知するまでの区間．(2)15 秒以内の連続した発話区間．話者分離は行なっていないため，1 発話に別の人の発話が含まれる可能性があることを考慮する．

4.1.2　評価用データ

我々の提案するタスクは，現在の発話文に関連する過去の会話ログを適合発話文書の集合 D_r を抽出することである．実際の利用シーンを考慮すると，過去の会話が必要になるシーンは，過去に明示的に固有名詞に対する説明が行われている場合が多い．固有名詞が出現しない暗黙的な関連も存在するが，明確に言及されていない分，提示しても不自由分なデータとなる可能性が多い．そのため，本実験では，固有名詞を中心に評価用データを作成する．

評価用データの作成方法を説明する．まず，検索対象のコレクションに対し想定される固有名詞（例えば,「ソシュール」「名古屋」）での検索を行う．これにより，2 回以上言及されている固有名詞を特定し，その特定の名詞が含まれる発話文集合を取り出す．この中で，最後に発話された文書の固有名詞以降を削除し，発話途中の入力文書 D_t とする．次に，発話途中の入力文書 D_t 以外の発話文書集合に対し，発話途中の入力文書 D_t との関連性判断を手動で行う．発話文書は一つの発話単位ごとに区切られているため，特定の名詞が含まれている発話文でも，部分的に関連する文書の中で特定の名詞が出現することがある．また，口頭での会話を書き起こしツールを用いて自動で書き起こしをする際に，異なる単語に誤認識することがある．例えば,「相槌」という発話を「愛知」と認識することがある．このように，ある特定の単語が含まれているが，部分的に関連するが，関連度が低い発話文や，全く関連性がない発話文が存在する．このことから，発話文集合のスコア付けを表 1 に基づいて行う．これにより，ある時点 t では，特定の名詞が出現していない発話文を用いて，過去にその名詞に言及した会話ログが抽出できるか評価するための評価用データを作成することができる．

4.2　データセットの統計情報

本実験では，スコア 1 のみを適合文書とし，最終的に 39 個の入力発話途中文書 D_t と適合文書集合 D_r のペアを得た．得られたデータセットは TREC 形式で保存した．表 2 に，作成したデータセットの統計量を示す．

4.3　実験設定

我々は提案手法の有意性や特性を示すために，次の手法を用いて実験をおこなった．

(1) BM25 発話文

これは本実験のベースラインとなる手法である．現在までの発話文を入力とし，クエリを作成し，検索を行う．

(2) BM25 予測発話文を用いたクエリ拡張

これは，発話文までの入力をもとに予測発話文の生成を行い，クエリを作成し，検索を行う．

(3) BM25 発話文 + 予測発話文を用いたクエリ拡張

これは，発話文までの入力をもとに，予測発話文の生成を行い，クエリを作成し，元の発話文から生成されたクエリを拡張し，検索を行う．

(2)，(3) 共に予測発話文の生成には今回は rinna 社が構築した日本語に特化した GPT-2 大規模言語モデルである japanese-gpt-1b を用いた．パラメータ数は 13 億である．また，形態素解析には sudachipy を用いた．本研究では，一つの入力に対し最短 100 語，最長 500 語のテキストが生成される設定としている．

構築したデータセットを用いた実験の手順を以下で述べる．ある時点 t での発話途中の文書 D_t とその文脈にあたる k 回にわたる発話文の集合 $D_i = \{d_{t-k}, d_{t-(k-1)}, ..., d_t\}$ を入力し，それぞれのモデルでの検索を行う．実験でのタスクでは，D_i と関連する文書 D_r のランキング問題とした．入力に用いた入力文書数は 39 個である．それぞれの入力に対して nDCG@k を計算し，平均 nDCG@k を各モデルの評価として用いた．

実験に用いたパラメータは以下の通りである．(1) 発話済み文章からクエリを作成する際に検索に用いる単語数である文脈窓幅 w は 7 語とした．これは，先行研究 [17] で文脈を多く入力するために文脈窓幅を大きくしすぎると精度が下がるといった結果が報告されており，適切な文脈窓幅である 7 語としている．(2) 拡張時に生成する予測発話文 $D_o = \{\hat{d}_1, \ldots, \hat{d}_l\}$ の数 l 個である．予測発話文の種類が増えるほど，ユーザの情報意図にあったクエリが生成される可能性が高いと考えられるため，さまざまな数値を利用してモデルの精度を検証する．実験でのタスクはある時点 t までの発話に対して関連発話文書を検索するランキング問題とした．

各モデルの性能評価は nDCG@k（$k = 5, 10, 100$）で行う．GPT-2 で生成される文書は確率論的に決まるため，5 回実験を行った平均をモデルの評価とする．また，必要に応じて，シス

表 3　各手法の精度（± 標準誤差）．生成文書数 $l = 10$．文脈窓の大きさ $w = 7$．手法間で最も精度が高い結果を太文字で示す．

手法	nDCG@k		
	k=5	k=10	k=100
（1）BM25	0.094 (0.000)	0.111 (0.000)	0.366 (0.000)
（2）BM25（予測発話文）	**0.131** (0.005)	**0.159** (0.005)	**0.444** (0.006)
（3）BM25（発話文+予測発話文）	0.116 (0.002)	0.131 (0.004)	0.414 (0.006)

テム間の統計的有意性を評価するために，$p = 0.05$ の対応のある t 検定を実行する．

4.4 実験結果

表 3 に，本研究で作成したデータセットでの各モデルの精度と標準誤差を示す．比較にはベースラインであるクエリ拡張をしない BM25(1) と，クエリ拡張ありの BM25(2)，(3) を用いた．また，検索時の生成文書数 l は 10 個，文脈窓の大きさ w を 7 語としたとする．予測発話文を用いたクエリ拡張による BM25 は全ての評価基準でベースラインを超えている．最も精度の高かった手法は，予測発話文のみをクエリ拡張に用いた BM25(2) であり，次点は，発話文と予測発話文どちらもクエリ拡張に用いた BM25(3) であった．nDCG@5 を元にすると，ベースライン手法とした発話文のみを用いた BM25(1) からの総合的な精度改善は 38% であった．また，統計的検定として，それぞれの提案手法とベースラインの間で $p = 0.05$ とした t 検定の両側検定を行なった．予測発話文を用いたクエリ拡張による BM25 の nDCG(0.049) は統計的に有意であると判断された．しかし，発話文と予測発話文を用いたクエリ拡張による BM25 の nDCG(0.078) は統計的に有意でないと判定された．この時, (　) の内の値は有意確率である．

次に，テキストの生成には計算機のコストと時間がかかる可能性があるため，クエリ拡張に必要なテキスト数を確認することは重要である．そのため，生成文書数のパラメータを変更し，どれだけ精度が変化をするかで評価を行った．その結果を表 4 に示す．(2) の提案手法はどの設定においてもベースライン手法を上回る精度を達成している．(3) の提案手法も生成文書数を増やすほど精度が向上していたが，(2) の方が劇的な精度の向上が見られる．また，(2) の提案手法は生成文書数が 2 個だったときの標準誤差が他のパラメータと比較すると大きい．つまり，生成数が少ないと生成文書の質によって精度がばらつくことため，文書数はある程度大きい方が良い．

図 2 を見ると，生成文書数を増やしてもある程度で nDCG@k の精度が頭打ちになることがわかる．これは，予測発話文を生成する過程で，不適合な文書を高く評価してしまうクエリが作られやすい文書が一定程度生成されているからであると考えられる．例えば,「あのいっこ僕最近気になったやつがあってあのコストコ」という入力文書があった場合,「BBQ」という単語が次に出現することになっているが，名詞が重なっているため，次の単語がうまく生成できず,「コストコ」という単語が繰り返し生成されることがあり，異常な検索クエリになることがある．これは，GPT-2 が，文脈から次に出現される単語を予測する生成モデルで，モデルの特性上，繰り返し同じ単語が出現する文書を生成してしまうことが理由に挙げられる．このことから，発話文から連続した単語を除外することや，文書の適合スコアの正規化などが必要になると考えられる．

表 4　各手法の nDCG@5（± 標準誤差）．それぞれ 5 回実行したときの平均をとっている．

生成文書数	手法		
	（1）BM25	（2）BM25（予測発話文）	（3）BM25（発話文+予測発話文）
0	0.094 (0.000)	0.094 (0.000)	0.094 (0.000)
2	0.094 (0.000)	0.116 (0.013)	0.116 (0.005)
5	0.094 (0.000)	0.129 (0.005)	0.113 (0.005)
7	0.094 (0.000)	0.134 (0.008)	0.112 (0.004)
10	0.094 (0.000)	0.131 (0.005)	0.116 (0.002)
15	0.094 (0.000)	0.138 (0.006)	0.122 (0.005)
17	0.094 (0.000)	0.134 (0.006)	0.118 (0.004)
20	0.094 (0.000)	0.137 (0.007)	0.117 (0.003)
50	0.094 (0.000)	0.132 (0.007)	0.121 (0.004)

5 まとめ

本論文では，現在の会話内容に関連する過去の発話内容を抽出し，円滑な会話を実現する問題に取り組んだ．この問題では，ユーザが掲示情報を理解し，会話に利用するために時間がかかってしまうため，明示的にトピックへの言及がなくとも関連会話データを抽出する必要があった．そこで，我々は発話文予測を用いたクエリ拡張という手法を提案した．

発話文予測を用いたクエリ拡張は，現時点までの発話内容を入力とし，GPT-2 を用いて生成された，入力以降に発話されそうな文書を用いて検索クエリを作ることで実現した．

提案手法の有意性を検証するために，複数回に渡り，過去の会話に言及しているデータセットが必要であったが，それに適したデータセットが入手できなかったため，新たにデータセットを作成した．データセットの作成には，Youtube 上で公開されている複数回に渡り会話を行なっている動画を用いた．実験

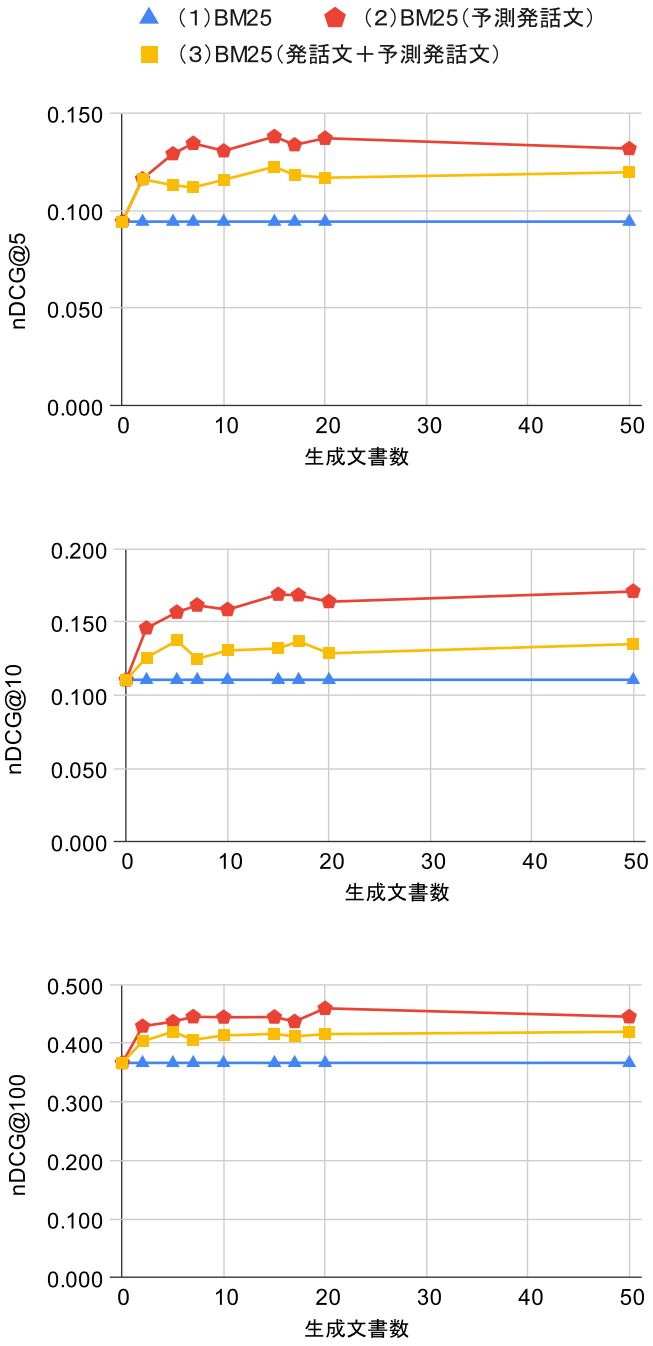

図 2　生成文書数を変化させたときの nDCG@k．それぞれ 5 回実行したときの平均をとっている．

では，このデータセットを用い，ベースライン手法と提案手法を比較し，提案手法の有意性を検証した．実験の結果，発話文予測を用いたクエリ拡張によってベースライン手法を上回る高い精度が得られることが明らかとなった．また，それぞれの手法の文書生成数と精度の間に関係があり，生成数を増やしても精度は頭打ちになることを示した．

今後の課題としては，発話文から連続した単語の除外や文書の適合スコアの正規化による検索結果の向上，検索結果の要約，実際の複数回にわたる自然会話，複数人での会話への拡張などを挙げる．

謝辞　本研究は JSPS 科研費 21H03554, 22H03905 の助成を受けたものです．ここに記して謝意を表します．

文　献

[1] Seyed Ali Bahrainian and Fabio Crestani. Are conversation logs useful sources for generating memory cues for recalling past memories? In *Proceedings of the 2nd Workshop on Lifelogging Tools and Applications*, pages 13–20, 2017.

[2] Seyed Ali Bahrainian and Fabio Crestani. Towards the next generation of personal assistants: systems that know when you forget. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*, pages 169–176, 2017.

[3] Matt-M Bouamrane and Saturnino Luz. Meeting browsing. *Multimedia Systems*, 12(4):439–457, 2007.

[4] Yi Chen and Gareth JF Jones. Augmenting human memory using personal lifelogs. In *Proceedings of the 1st augmented human international conference*, pages 1–9, 2010.

[5] John Choi, Donald Hindle, Fernando Pereira, Amit Singhal, and Steve Whittaker. Spoken content-based audio navigation (scan). In *Proceedings of the ICPhS*, volume 99, 1999.

[6] Vincent Claveau. Query expansion with artificially generated texts. *arXiv preprint arXiv:2012.08787*, 2020.

[7] Xinya Du, Luheng He, Qi Li, Dian Yu, Panupong Pasupat, and Yuan Zhang. Qa-driven zero-shot slot filling with weak supervision pretraining. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 654–664, 2021.

[8] John S Garofolo, Cedric GP Auzanne, Ellen M Voorhees, et al. The trec spoken document retrieval track: A success story. *NIST SPECIAL PUBLICATION SP*, 500(246):107–130, 2000.

[9] Anjuli Kannan, Kai Chen, Diana Jaunzeikare, and Alvin Rajkomar. Semi-supervised learning for information extraction from dialogue. In *Interspeech*, pages 2077–2081, 2018.

[10] Nan Li, Frédéric Kaplan, Omar Mubin, and Pierre Dillenbourg. Supporting opportunistic search in meetings with tangible tabletop. In *CHI'12 Extended Abstracts on Human Factors in Computing Systems*, pages 2567–2572. 2012.

[11] Jun Liu, Tong Ruan, Haofen Wang, and Huanhuan Zhang. Prompt-based generative approach towards multi-hierarchical medical dialogue state tracking. *arXiv preprint arXiv:2203.09946*, 2022.

[12] Samuel Louvan and Bernardo Magnini. Recent neural methods on slot filling and intent classification for task-oriented dialogue systems: A survey. *arXiv preprint arXiv:2011.00564*, 2020.

[13] Mayur Patidar, Puneet Agarwal, Lovekesh Vig, and Gautam Shroff. Automatic conversational helpdesk solution using seq2seq and slot-filling models. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1967–1975, 2018.

[14] Andrei Popescu-Belis, Jonathan Kilgour, Peter Poller, Alexandre Nanchen, Erik Boertjes, and Joost de Wit. Automatic content linking: speech-based just-in-time retrieval for multimedia archives. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 703–703, 2010.

[15] Kunpeng Qin, Harrisen Scells, and Guido Zuccon. Pecan: A platform for searching chat conversations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2610–2614, 2021.

[16] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsu-

pervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[17] Sosuke Shiga, Hideo Joho, Roi Blanco, Johanne R Trippas, and Mark Sanderson. Modelling information needs in collaborative search conversations. In *Proceedings of the 40th international acm sigir conference on research and development in information retrieval*, pages 715–724, 2017.

[18] Liqiang Song, Mengqiu Yao, Ye Bi, Zhenyu Wu, Jianming Wang, Jing Xiao, Juan Wen, and Xin Yu. Ls-dst: Long and sparse dialogue state tracking with smart history collector in insurance marketing. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1960–1964, 2021.

[19] Nikos Voskarides, Dan Li, Pengjie Ren, Evangelos Kanoulas, and Maarten de Rijke. Query resolution for conversational search with limited supervision. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 921–930, 2020.

[20] Alex Waibel, Michael Bett, Michael Finke, and Rainer Stiefelhagen. Meeting browser: Tracking and summarizing meetings. In *Proceedings of the DARPA broadcast news workshop*, pages 281–286, 1998.

[21] 渡邉涼太. 自動検索のためのクエリ生成手法の開発と分析. In **筑波大学 情報学群 知識情報・図書館学類** *2019* **年度 卒業論文**, 2020.

[22] 野口敏. **誰とでも** *15* **分以上 会話が途切れない！ 話し方 そのまま話せる！ お手本ルール** *50*. 株式会社すばる社, 東京, 2013.