

修士論文

特許検索における質問意図の曖昧化

48-156229 胡 瀚林

指導教員 中川 裕志 教授

2017 年 1 月

東京大学大学院情報理工学系研究科数理情報学専攻

概要

企業が特許を取る前に、類似の特許が既に存在するかを確かめるために特許データベースを検索する必要がある。しかし、検索の質問から企業秘密が漏洩する可能性がある。ウェブテキスト検索の質問から質問者の検索意図を守る手法が多数存在している。その中では真の質問と同時にダミー質問を提出する質問曖昧化手法が一番効率的、現実的である。本論文では特許検索における既存の質問曖昧化手法を実装し、類似度攻撃で特許データベースにおける既存手法の安全性を評価した。

また、類似度攻撃を含め、多くの既存の質問曖昧化に対する攻撃手法は攻撃者が質問者に関する事前情報を持つと仮定する。本論文では事前情報なしの攻撃手法を提案し、その攻撃手法に対応する既存の質問曖昧化の改良と新たな質問曖昧化手法を提案する。

目次

第 1 章	はじめに	1
第 2 章	特許の概要	2
2.1	特許分類	2
2.2	特許検索	5
第 3 章	曖昧化検索	6
3.1	否認可能検索	7
3.2	質問者のプライバシーを保護する質問加工法	10
3.3	質問意図を曖昧化するキーワード検索	15
第 4 章	意味分析	17
4.1	tf-idf	17
4.2	潜在意味分析	18
4.3	潜在ディリクレ配置法	18
第 5 章	攻撃手法	20
5.1	メイントピック攻撃	20
5.2	事前情報がある場合の類似度攻撃	21
5.3	事前情報がない場合の類似度攻撃	21
第 6 章	単語ベクトルを用いた質問曖昧化	23
6.1	単語ベクトル	23
6.2	質問者が検索したいトピックを曖昧化する質問曖昧化	24
6.3	質問者が検索したいトピックにおける質問曖昧化	25
6.4	データベース分割	25
第 7 章	評価実験	27
7.1	文書集合と質問集合	27
7.2	メイントピック攻撃	28
7.3	事前情報がない場合の類似攻撃	31

iv 目次

7.4	事前情報がある場合の類似攻撃	32
7.5	データベース分割	32
7.6	交差攻撃	33
7.7	検索曖昧化手法の比較	34
第 8 章	おわりに	36
	謝辞	37
	参考文献	38

第 1 章

はじめに

テキスト検索をするとき、検索質問をサーバ側に渡さなければならない。しかし、検索質問から質問者の情報が漏洩する危険があることが AOL 事件 [Michael and Tom, 2006] より証明された。特許検索の場合は検索質問が研究開発動向など企業秘密を含んでいるため、一般的なウェブ検索の質問者より質問のプライバシー問題を重視している。そのような問題を解く様々な手法が存在している。[Dingledine et al., 2004] や [Saint-Jean et al., 2007] などの IP アドレスの匿名化メカニズムは登録情報が必要な検索サーバに対応できない。また検索質問のみから質問者を一意に特定されてしまう可能性がある。プライベート情報検索 (Private Information Retrieval)[Chor et al., 1998] は計算量的安全性を持つが、サーバ側で大量の計算が必要であるため実用化することは困難である。曖昧化検索 (Obfuscation Search)[Balsa et al., 2012] は真の質問を分析し適切な $K - 1$ 個のダミー質問を生成し真の質問と同時に検索する。安全性が弱いですが、効率よく質問者の検索意図を守ることができる。

本論文の構成は次の通りである。第 2 章では特許文書と特許検索の特徴を述べる。第 3 章では既存の質問曖昧化メカニズム [Murugesan and Clifton, 2009, Pang et al., 2010, Wang and Ravishankar, 2014] を述べる。第 4 章では曖昧化メカニズムがよく用いる意味分析手法を述べる。第 5 章では既存の攻撃手法 [Petit et al., 2016] を述べ、[Petit et al., 2016] の改良と新たな攻撃手法を提案する。第 6 章では新たな質問曖昧化手法を提案する。最後に、第 7 章で評価実験を述べ、第 8 章で全体をまとめる。

第 2 章

特許の概要

特許検索質問のプライバシーを保護する手法を説明する前に特許検索と特許そのものを簡単に紹介する必要がある。特許法第 1 条には、「この法律は、発明の保護及び利用を図ることにより、発明を奨励し、もつて産業の発達に寄与することを目的とする」とある。特許制度は、発明者には一定期間、一定の条件のもとに特許権という独占的な権利を与えて発明の保護を図る一方、その発明を公開して利用を図ることにより新しい技術を人類共通の財産としていくことを定めて、これにより技術の進歩を促進し、産業の発達に寄与しようというものである。特許を取るには以下の条件を満たさなければならない：

1. (新規性：特許法 29 条第 1 項) 特許出願前に公然知られた発明、公然実施をされた発明、頒布された刊行物に記載された発明又は電気通信回線を通じて公衆に利用可能となった発明について特許を受けることができない。
2. (進歩性：特許法 29 条第 2 項) 特許出願前にその発明の属する技術の分野における通常の知識を有する者が前項各号に掲げる発明に基いて容易に発明をすることができたときは、その発明については、同項の規定にかかわらず、特許を受けることができない。

すなわち、特許を出願する前に既存の特許を検索し、自分の発明について新規性と進歩性の有無を判断する必要がある。また、特許を受けようとする新規性と進歩性がある発明を特定できる特許請求の範囲を記載する必要がある。

図 2.1、2.2 は特許文書の例である。発明の範囲を正確に記載するように請求項は普段に使わない学術用語を用いる。また、一般的な文書は単語をなるべく重複しないようにする一方、特許文書は単語を全体を通じて統一して使用し、指示代名詞はなるべく用いない。そのため、特許データベースでは一般的なウェブ文書データベースより多くの単語があり、単語の曖昧性が少ない。

2.1 特許分類

特許では人手によって分類され、特定の分類コードが付いている。今、最も使われている特許分類が世界知的所有権機関 (WIPO) によって管理されている国際特許分類 (IPC) である。

JP 2016-208844 A 2016.12.15

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号
特開2016-208844
(P2016-208844A)

(43) 公開日 平成28年12月15日 (2016.12.15)

(51) Int.Cl.
A01B 35/04 (2006.01)

F1
A01B 35/04
A01B 35/04

テーマコード (参考)
2B034
E
B

審査請求 未請求 請求項の数 4 O L (全 13 頁)

(21) 出願番号	特願2015-92186 (P2015-92186)	(71) 出願人	390010836
(22) 出願日	平成27年4月28日 (2015.4.28)		小橋工業株式会社
		(74) 代理人	110000408
			特許業務法人高橋・林アンドパートナーズ
		(72) 発明者	河原 文雄
			岡山県岡山市南区中畦684番地 小橋工業株式会社内
		Fターム(参考)	2B034 AA03 BA06 BB01 BB02 EA02 EB06 EB33 JA06

(54) 【発明の名称】 農作業機

(57) 【要約】

【課題】 代かき作業機を昇降させる必要がない場面において、オート装置が代かき作業機を昇降させることを防止する。

【解決手段】 本発明の一実施形態に係る農作業機は、耕耘作業を行うロータリ作業部を回転自在に支持する機体と、機体に設けられ、ロータリ作業部の上部を覆うカバー部と、カバー部の後端部に回転可能に支持されたエプロンと、エプロンの背面に取り付けられた支持部材と、カバー部に取り付けられ、エプロンのロック状態とフリー状態を切り替え可能なエプロン回動制御部と、を備え、エプロン回動制御部は、後端部が支持部材に対し取り付けられたロッド部と、ロッド部の前端部を回転自在に支持し、被係合部を有する第1アーム部と、カバー部に回転自在に支持され、係合部を有する第2アーム部と、第2アーム部を回動させる駆動部とを有し、係合部は、第1アーム部が回動するときに、被係合部の回動を規制するように構成されてもよい。

【選択図】 図1

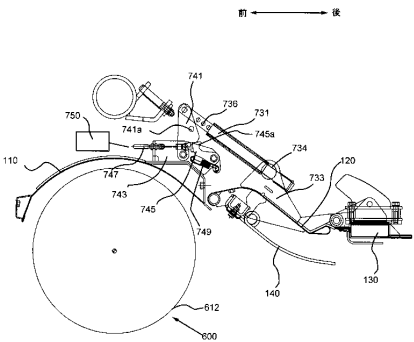


図 2.1. 特許文書例 (a)

4 第2章 特許の概要

(2)

JP 2016-208844 A 2016.12.15

【特許請求の範囲】

【請求項1】

耕耘作業を行うロータリ作業部を回転自在に支持する機体と、
前記機体に設けられ、前記ロータリ作業部の上部を覆うカバー部と、
前記カバー部の後端部に回動可能に支持されたエプロンと、
前記エプロンの背面に取り付けられた支持部材と、
前記カバー部に取り付けられ、前記エプロンが自由に回動できない状態であるロック状態と前記エプロンが自由に回動できる状態であるフリー状態とに切り替えることができるエプロン回動制御部と、を備え、

前記エプロン回動制御部は、後端部が前記支持部材に対して、摺動可能に取り付けられたロッド部と、

前記ロッド部の前端部を回動自在に支持し、前記カバー部に対して回動自在に支持されて、被係合部を有する第1アーム部と、

前記カバー部に回動自在に支持され、係合部を有する第2アーム部と、

前記第2アーム部を回動させる駆動部とを有し、

前記係合部は、前記ロッド部が前方に移動するのに伴い前記第1アーム部が回動するときに、前記被係合部の回動を規制することを特徴とする農作業機。

【請求項2】

前記被係合部は、ピン部材であることを特徴とする請求項1に記載の農作業機。

【請求項3】

前記駆動部は、ワイヤとワイヤ制御部を含むことを特徴とする請求項1又は請求項2に記載の農作業機。

【請求項4】

耕耘作業を行うロータリ作業部を回転自在に支持する機体と、

前記機体に設けられ、前記ロータリ作業部の上部を覆うカバー部と、

前記カバー部の後端部に回動可能に支持されたエプロンと、

前記エプロンの背面に取り付けられた支持部材と、

前記カバー部に取り付けられ、前記エプロンが自由に回動できない状態であるロック状態と前記エプロンが自由に回動できる状態であるフリー状態とに切り替えることができるエプロン回動制御部と、を備え、

前記エプロン回動制御部は、後端部が前記支持部材に対して、摺動可能に取り付けられたロッド部と、

前記ロッド部の前端部を回動自在に支持し、前記カバー部に対して回動自在に支持されて、係合部を有する第1アーム部と、

前記カバー部に回動自在に支持され、被係合部を有する第2アーム部と、

前記第2アーム部を回動させる駆動部とを有し、

前記被係合部は、前記ロッド部が前方に移動するのに伴い前記第1アーム部が回動するときに、前記係合部の回動を規制することを特徴とする農作業機。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、農作業機に関する。特に、本発明は、エプロンが自由に回動できない状態であるロック状態とエプロンが自由に回動できる状態であるフリー状態とに切り替えることができるエプロン回動制御部を備える農作業機に関する。

【背景技術】

【0002】

耕耘ロータにより耕耘された耕土を整地するエプロン（第1整地板）とエプロンの後部に上下方向に回動自在に設けられて耕土表面を均平にするレベラ（第2整地板）を備える農作業機、例えば、代かき作業機は、一般に、走行可能な走行機体の後部に三点リンク連結機構を介して昇降可能に連結されて、走行機体の前進走行とともに進行しながら代かき

図 2.2. 特許文書例 (b)

IPC は世界標準であるため，同じ分類コードが付いているどの国の特許も同じ分類に属する．国際特許分類は階層構造であり，一番上の階層は A から H までの 8 個のセクションである．セクション以下は表 2.1 に表したように四つの階層に分類されている．

セクション:A	健康および娯楽
サブセクション: 61	医学または獣医学;衛生学
クラス: C	歯科:口腔または歯科衛生
メイングループ:5	歯の充填または被覆
サブグループ:08	歯冠:その製造; 口中での歯冠固定

表 2.1. 国際特許分類例:A61C 5/08

一般に，1 つの特許が複数の分類に属する．本論文で用いる特許データベースでは一件あたり平均 2.4 個の分類コードが付いている．本論文では特許発明の主体を表す筆頭コードをその特許が属する分類とする．

2.2 特許検索

特許データベースにおける検索は技術水準調査，新規性調査，無効資料調査と侵害調査に分けられる．表 2.2 では 4 つの検索タイプを検索対象あるいは検索質問の発端となるものと検索の目的を示す．本論文では特許の新規性調査における検索質問の安全性について分析する．新規性調査はまだ出願していない発明について検索するため，質問意図が漏れたら，自社の研究開発動向など企業秘密が知られ，攻撃者に先に出願される恐れがある．

検索タイプ	検索対象 (specification)	検索目的
技術水準調査 (State of the Art Search)	アイデア	自分の発明に関連する背景知識を得る
新規性調査 (Novelty Search)	特許願書	特許登録の可能性を判断する
無効資料調査 (invalidity Search)	特定な特許	発明の特許性の有無を判断する
侵害調査 (Infringement Search)	商品と 商品に関連する技術	権利侵害とならないかを判断する

表 2.2. 特許検索タイプ

本論文では NTCIR-6[Fujii et al., 2007] で用いた特許文書集合と無効資料検索タスクの質問を用いて評価実験をする．無効資料調査の目的は第三者の発明に特許性がないことを示す根拠となる文書を検索することであるが，特定発明の特許性の有無を判断することは新規性調査と一致するため，無効資料検索タスクの質問を新規性調査の質問とすることが妥当であると考えられる．

第 3 章

曖昧化検索

曖昧化検索は質問者が検索したい真の質問と質問者側で生成したダミー質問を 1 つの質問グループにし、検索サーバに提出し、真の質問がどれかを曖昧化するものである。本論文では以下のモデル [Balsa et al., 2012] を用いて既存の曖昧化検索メカニズムを分析する。質問者 Alice がとある検索サーバに質問を入力して手に入れたい情報を検索し、検索サーバが semi-honest な攻撃者であることを仮定する。Semi-honest な攻撃者はプロトコルには従うが、必要以上の情報を得ようと試みる。

質問が単語の集合であり、質問の定義域を単語集合の冪集合にする。

定義 1. ユニバーサル質問集合 Q : W を全ての単語の集合とする。ユニバーサル質問集合 Q とは W の冪集合である、つまり

$$Q = P(W) = \{A | A \subset W\} \quad (3.1)$$

Alice のプロフィールを多項分布と仮定し、Alice が持つ真のプロフィールを X とする。

定義 2. 質問者のプロフィール X : T を全てのトピックの集合とする。質問者のプロフィール X とは

$$X = \{x_i | i \in T\} \quad (3.2)$$

x_i は質問者がトピック i に対して持つ興味の強さを表す。

曖昧化検索メカニズムは Alice のコンピュータで実行する。曖昧化検索メカニズムが意味分析ツール SA を用いて真の質問 q_R を分析しダミー質問 q_D を生成する。生成したダミー質問 q_D と真の質問 q_R を 1 つの質問グループにし、検索サーバに提出する。意味分析ツール SA を用いるとき質問 q とトピック t の関係を表す関数は以下のように定義する、

定義 3. 質問-トピックスコア関数: r_{score}_{SA} : T を全てのトピックの集合とする。質問 q とトピック t の関係を表す関数とは

$$r_{score}_{SA} : Q \times T \rightarrow \mathbb{R} \quad (3.3)$$

定義 4. 質問 q のメイントピック: $\delta_{SA}(q)$: T を全てのトピックの集合とする。質問 q のメイ

ントピック $\delta_{SA}(q)$ とは

$$\delta_{SA}(q) = \operatorname{argmax}_{t \in T} rscore_{SA}(q, t) \quad (3.4)$$

次に質問 q のトピックベクトルを定義する．質問 q のトピックベクトル $tvec_{SA}(q) = (rscore_{SA}(q, t_1), \dots, rscore_{SA}(q, t_{|T|}))$ とは q と全てのトピック t_i の質問-トピックスコア関数 $rscore(q, t_i)$ を要素として持つ $|T|$ 次元ベクトルである．質問のトピックベクトルを使って質問間の関係を評価することができる．

検索サーバが Alice からもらった質問をすべて記録し，その全質問を分析して得るプロフィールを Y とする．

定義 5. 質問比較関数: C . 質問比較関数 $C : Q \times Q \rightarrow \mathbb{R}$ を以下のように定義する

$$C_{SA}(q_1, q_2) = \frac{(tvec_{SA}(q_1) \cdot tvec_{SA}(q_2))}{\|q_1\| \|q_2\|} \quad (3.5)$$

曖昧化検索は 3 つ違うレベルの目標がある．まずは質問そのものの曖昧化である．質問者が検索した真の質問 q_R はどの質問であるかをわからないようにする．2 つ目は質問意図の曖昧化である．質問者が検索したいものは何であるかをわからないようにする．最後は質問者のプロフィール X の曖昧化である． Y から質問者が興味を持つトピックは何であるかをわからないようにする．

質問の曖昧化ができたとしても質問意図の曖昧化ができると限らない．林檎とリンゴの 2 つ質問から真の質問を確定することができないが，質問者が林檎について検索したいことが確定できる．同じように林檎と梨の 2 つ質問から質問者が検索したいを確定することができないが，質問者が果物に興味を持つことが確定できる．本論文では質問意図の曖昧化をメインにする．

次に検索質問のプライバシー保護の代表的な手法，否認可能検索 (PDS)[Murugesan and Clifton, 2009]，質問者のプライバシーを保護する質問加工法 (ETSQ)[Pang et al., 2010]，質問意図を曖昧化するキーワード検索 (HDGA)[Wang and Ravishankar, 2014] を紹介する．

3.1 否認可能検索

否認可能検索という概念を提出したのは [Murugesan and Clifton, 2008] である．つまり，サーバは特定な質問者が特定の時間に提出した一連の質問 $L = q_1, q_2, \dots, q_K$ のログを持つと仮定する．ログにアクセスしたある人が真の検索質問が q_i だと証明したいとき， L の中の任意の質問 q_j が真の質問となる確率が等しく $1/K$ だと証明できる．以下に否認可能検索を定義する．

定義 6. k -否認可能検索：質問 q を質問者が入力した質問とする．ダミー質問生成システム DGS が k 個の質問を含んでいる質問集合 $DGS(q_u) = \{q_1, \dots, q_k\}$ を出力しサーバに提出する． $DGS(q_u)$ が以下の性質を持つなら， $DGS(q_u)$ を PD-質問集合といい， D を k -否認可能検索という

1. $\exists q_i \in DGS(q_u), q_i$ と q_u が意味的に近い
2. $\forall q_j \in DGS(q_u), DGS(q_j) = DGS(q_u)$
3. $\forall q_j \in DGS(q_u), q_j$ が違うトピックに含まれる
4. $\forall q_j \in DGS(q_u), q_j$ が同じような尤もらしさを持つ

PDS では事前に文書集合から高頻度な単語と単語ペアをシード質問として抽出し、潜在意味分析 (LSA)[Deerwester et al., 1990] を用いてシード質問をトピック空間にマップし、トピック空間に距離が近いシード質問をクラスタリングして標準質問と PD-質問集合を構築する。検索する場合は、質問者が検索したい質問の代わりに事前に用意した標準質問集合からトピック空間において質問者が検索したい真の質問と最も近い標準質問が属する PD-質問集合をサーバに提出する。次にサーバから検索結果を得て、質問者側で真の質問を用いて検索結果をフィルタリングする。以下でこの流れを具体的に述べる。

3.1.1 シード単語と標準質問

システムが生成した質問は通常は使わない単語の組み合わせを使うことがある。攻撃者がこのような質問をダミー質問と判定し、真の質問を特定する可能性があるため、PDS は標準質問と PD-質問集合を事前に構築する。そのため、 Q 中の全ての質問をカバーすることは不可能である。PDS の目標は妥当な再現率を得ることであるため、高頻度な単語だけを使うことは適当だと考えられる。

Algorithm 1 標準質問の構築

Input: シード質問集合 S

- 1: $Q_C \leftarrow \phi$
- 2: Kdtree を構築し S の全ての要素を追加する
- 3: **for all** $s_i \in S$ **do**
- 4: Kdtree を用いて s_i と最も近いシード質問 c_1, c_2 を選ぶ
- 5: $cquery = s_i \cup c_1 \cup c_2$
- 6: **if** $cquery \notin Q_C$ **then**
- 7: $Q_C = Q_C \cup \{cquery\}$

Output: 標準質問の集合 Q_C

まず、単語・文書行列に頻出パターンマイニング [Bodon, 2010] を用いて Δ 回以上に表れた単語と連続する単語からなる単語ペアをシード質問として抽出し、トピック空間にマップする。シード質問は質問者の意図を適切に表さないことが多いため、PDS では意味的に近いシード質問をグループにして標準質問にする。アルゴリズム 1 ではこの流れを具体的に説明する。このステップの計算量は $O(N \log N)$ となる。ここで N はシード質問の数である。

3.1.2 PD-質問集合の構築

PD-質問集合を構築するには、トピックは異なるが尤もらしさが近い標準質問を同じ質問集合に集めれば良い。そのため、多様性と尤もらしさを計算する方法を提案する必要がある。多様性ではトピック空間の中の距離で評価する。人間が作った質問と比較するため、合理的な大きさを持つ質問ログ $Q_L = \{q : q \in Q\}$ にアクセスできると仮定する。 Q_C と同様に Q_L もトピック空間にマップし、標準質問の近傍の中の Q_L の要素数で標準質問の尤もらしさを計算する。近傍に多くの Q_L に含まれる質問がある標準質問を尤もらしさが高いとする。

次に3つの部分の和となる標準質問間の関係を評価する関数を定義する。質問 q_1 と質問 q_2 のユークリッド距離 $edist(q_1, q_2)$ とは、

$$edist(q_1, q_2) = \sqrt{\sum_{i \in T} (tvec_{LSA}(q_1)[i] - tvec_{LSA}(q_2)[i])^2} \quad (3.6)$$

である。ユークリッド距離が大きい質問が異なるトピックに含まれると考えられる。質問 q の強度とは、

$$\|q\| = \sqrt{\sum_{i \in T} (tvec_{LSA}(q)[i])^2} \quad (3.7)$$

である。質問 q の近傍中の質問数 $nhc(q)$ とは、

$$nhc(q) = count(tvec_{LSA}(q), Q_L, HCUBE(tvec_{LSA}(q), \vec{(\delta)})) \quad (3.8)$$

である。ここで、 Q_L は質問ログ、 $HCUBE(tvec_{LSA}(q), \vec{(\delta)})$ は $tvec_{LSA}(q)[i] \pm \delta[i]$ となる超立方体であり、 $count$ は超立方体中で Q_L に属するベクトルの数を返す。

定義 7. 質問間の評価関数: dis .

$$dis(q_1, q_2) = (1 - \frac{edist(q_1, q_2)}{\alpha}) + \frac{||q_1|| - ||q_2||}{\beta} + \frac{|nhc(q_1) - nhc(q_2)|}{\gamma} \quad (3.9)$$

ここで、 α は Q_C に属する全ての質問ペア間の最大のユークリッド距離、 β は質問ペア間の最大の強度差で、 γ は質問ペア間の最大の近傍中の質問数の差である。

したがって、近傍中の質問数と強度の差が小さく、トピック空間中の距離が大きい質問ペアの評価関数の値が低くなり、一つの PD-質問集合に入れるべきである。

次では、質問集合間の評価関数を定義する。 $A = a_1, \dots, a_n$ と $B = b_1, \dots, b_m$ を2つ質問集合とする。 A, B 間の評価関数とは、

$$dis(A, B) = (1 - \alpha_1/\alpha) + \beta_1/\beta + \gamma_1/\gamma \quad (3.10)$$

である。ここで、 $\alpha_1 = \min_{i,j} (edist(a_i, b_j))$ は2つの質問集合に属する質問ペア間のユークリッド距離の最小値であり、 $\beta_1 = |\frac{\sum_i ||a_i||}{n} - \frac{\sum_j ||b_j||}{m}|$ と $\gamma_1 = |\frac{\sum_i nhc(a_i)}{n} - \frac{\sum_j nhc(b_j)}{m}|$ は質問集合の強度と近傍中の質問数の平均の差である。

3.1.3 凝集型クラスタリング

PDS では、まず質問ペアを要素とするレベル 1 集合 L_1 を生成する。 Q_C に属する全の質問ペア間の評価関数の値の行列を計算し、評価関数の値が小さい順で質問ペアを L_1 に加える。質問ペア (q_i, q_j) に対し、 q_i か q_j は評価関数の値がもっと小さいペアに属する可能性がある。その場合、 q_i か q_j がすでに L_1 にあることとなり、次に評価関数の値が小さい質問ペアを選ぶ。選んだ質問ペアをマージし、次のレベルの集合 (L_2, L_3, \dots) を作る。マージステップはレベル変数 l が $\log_2 k$ になるまで続ける。したがって、最終レベルの集合中の質問クラスターの大きさが k となり、オーバーラップがないと保証する。

3.1.4 PD-質問集合の使用

真の質問 q_u に近い標準質問を探すため、 q_u を LSA を用いてトピックベクトル $tvec_{LSA}(q_u)$ にし、質問比較関数 $C(q_u, q_c)$ が一番大きい標準質問 q_c を選び、 q_c が属する PD-質問集合をサーバに提出し、質問者側で真の質問を用いて検索結果をフィルターする。一定の再現率を得るため、普段の検索より多くの文書を手に入れる必要があるが、フィルターステップがこの影響をなくす。

真の質問の全ての単語が PD-質問集合を構築するために使った単語リストに含まれていないなら、真の質問を LSA を用いてトピックベクトルにすることは不可能である。しかし、単語量が十分大きいなら、そのような状況が発生する可能性は低いと考えられる。また、(十分大きな) 単語リストに含まれてない単語は質問者の意図を漏洩するリスクが高い。質問者がそのような質問を提出したとき、質問者に危険性を警告し、検索しないようにすることが考えられる。

3.2 質問者のプライバシーを保護する質問加工法

今、テキスト検索サーバの大半が類似検索である。全ての質問単語を含んでいる文書しか検索できないキーワード検索と違い、類似検索は文書と質問の関連性を計算し文書にスコアをつける [Zobel and Moffat, 2006]。毎回全ての文書との関連値を計算しないために、検索サーバが単語と文書の類似度を転置ファイルに保存し、質問の単語と文書の類似度の和を質問とその文書の関連値とする。このような計算が必要であるため、[Bethencourt et al., 2006, Freedman et al., 2005, Boneh et al., 2004, Song et al., 2000] などキーワード検索しか対応できない研究は類似検索に応用できない。

PDS をはじめに多くの曖昧化検索メカニズム [Wang and Ravishankar, 2014, Shapira et al., 2005, Josep Domingo Ferrer et al., 2009] は質問の全体を分析し、適切な $K - 1$ 個のダミー質問を選ぶ。質問の全体ではなく単語ごとにダミー単語を混ぜれば、真の質問である可能性がある質問数が増え、攻撃者が真の質問を見破る確率が下がる。質問者が 1 つのトピックに対して検索するとき、1 つの単語を複数回使うと考えられる。毎回違うダミー単語を混ぜ

と同じ質問者の質問に出現する頻度が高い単語が真の質問単語となる可能性が頻度が低い単語より大きくなる．そうしたリスクを防ぐため ETSQ は単語バケットを事前に作り，真の質問単語と同じバケットにある他の単語をダミー単語とする．また，単語ごとにダミーを混ぜるため長い質問と類似検索に対応できる．

3.2.1 類似検索

コーパス D における検索サーバが質問を処理するとき基本的には転置ファイルを用いている．転置ファイルは質問単語の集合 W と全ての単語の転置リストからなる．単語 $w_i \in W$ の転置リスト L_i が $\langle d_i, p_{ij} \rangle$ の列である． $p_{ij} \in \mathbb{R}$ は単語 w_i と文章 $d_i \in D$ の関連値である． w_i が d_i に現れたなら p_{ij} の値は 0 より大きい，現れなかったなら 0 となる．空間圧縮のために $p_{ij} = 0$ な d_i は L_i に含まれていない．

質問 $q = \{w_i\}$ と文書 d_i と関連値は以下のように計算する

$$Score_{d_j, q} = \sum_{w_i \in q} p_{ij} \quad (3.11)$$

したがって，転置リスト L_i に含まれている文書だけが 0 より大きいスコアを持ち， q と関連があると見なす．転置ファイルを全体暗号化しても，サーバは転置リストの長さとアクセス頻度などの情報から真の関係値を推定できるため，そのような方法は無意味だと考えられる．

3.2.2 単語バケット

単語バケットを作るには 2 つ主要なリスクがある．真の質問単語が全て同じトピックについて述べると考えられる．そのような単語がランダムに選んだダミー単語と区別することが簡単である．また特許検索に多く使っている専門用語など特殊な単語と一般的よく使っている単語を混ぜると，専門用語が真の質問である可能性が大きいと考えられる．そんなリスクを減らすために，以下の特徴を持つ単語バケットを作りたい：

- (i) 同じバケットにある単語の特殊さは近いが，意味的には大きい違いがある．
- (ii) 2 つのバケットの全ての単語間の意味的距離の差が小さい．

検索するとき，質問単語が属するバケット中の他の単語がダミー単語として質問に加える．したがって，特殊な単語のダミー単語がいつも同じような特殊さを持ち，ダミー単語間の関係と真の質問の単語間の関係が似ていると考えられる．

紹介文献では単語を類義関係のセット (synset) でグループ化し，一つの synset が一つの概念に対応し，各 synset は上位下位関係，全体部分関係などの関係でリンクされている WordNet[Miller, 1995] を用いて単語バケットを作る．

2 つの単語が属する synset 間の最短パスを単語間の意味的距離とする．また上位下位関係でリンクされた 2 つ synset の中下位語が上位語より特殊であると考えられる．WordNet の中で実体 (entity) 以外全部の名詞 synset の上位語が唯一に存在する．上下位関係を枝とすると，

WordNet 中の名詞 synset が実体を根とする木となる。単語が属する一番深さが浅い synset の深さを単語の特殊レベルとし、レベルが大きければ大きいほど単語が特殊である。

3.2.3 バケット作り

本節では単語バケットを作る方法を述べる。まず、アルゴリズム 2 を用いて WordNet データベース中の意味的に近い単語を隣にして全ての単語一列に並べる。リンクが多い synset が意味的に豊富であるため、単語を一列に繋がる種として使われ、synset の関係数が大きい方から小さい方への順で処理する。複数の意味を持つ単語が属する各 synset が意味的に近いと考えられ、同じ単語を持つ synset を隣に並べる。また反意関係、上位下位関係、全体部分関係を持つ synset を隣に並べる。2 つの操作により、列に近い単語の意味も近いと保証する。

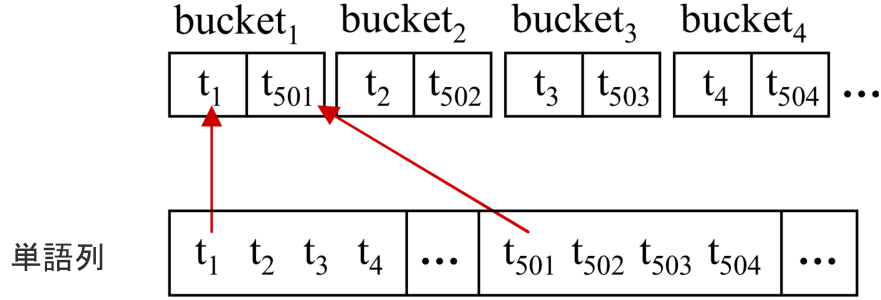
WordNet データベースにアルゴリズム 1 を行った結果データベース中全ての 117,798 個の名詞を一列に並べた。

Algorithm 2 単語を一列に並べる

```

1: function PROCESSSYNSET(synset ss)
2:   if  $ss$  の単語が複数の既存の単語列に含まれている then
3:     そんな単語列を結合する
4:     結合した単語列を  $sq$  にする
5:   else if  $ss$  の単語が既存の単語列に含まれていない then
6:     新たな単語列を作る
7:   else  $ss$  の単語の一つが一つ既存の単語列に含まれている
8:     その単語列を  $sq$  にする
9:   処理していない  $ss$  の単語を  $sq$  に加える
10:   $ss$  の単語を処理したとマークする
11:   $ss$  を処理したとマークする
12:  単語列  $sq$  を返す
13: function SEQUENCEVOCAB(WordNet wndb)
14:  全ての synset を関係数が多い方から小さい方への順で並べる
15:  全ての synset を処理していないとマークする
16:  全ての単語を処理していないとマークする
17:   $SeqSet = \phi$ 
18:  for all 処理していない synset  $ss$  do
19:     $sq = ProcessSynset(ss)$ ;  $sq$  を  $SeqSet$  に加える
20:    for all  $ss$  と反意関係、上位下位関係、全体部分関係をもつ synset  $ss'$  do
21:      処理していない  $ss'$  の単語を  $sq$  に加える
22:       $ss'$  の単語を処理したとマークする
23:       $sq = ProcessSynset(ss')$ ;  $sq$  を  $SeqSet$  に加える

```

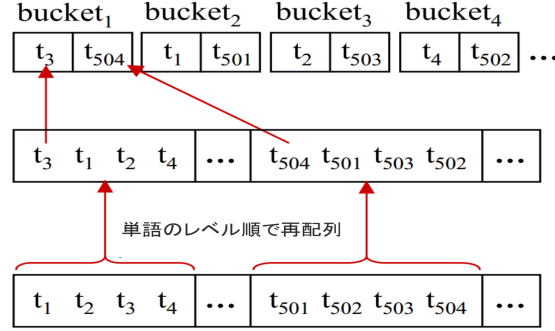
図 3.1. バケット作り- $N = 1000, BktSz = 2$ **Algorithm 3** 単語列から単語バケットを作る

```

1: function GENERATEBUCKETS(sq,BktSz,Segsz)
2:    $N =$  単語列  $sq$  の長さ
3:    $\#Seg = N/SegSz$ 
4:    $sq$  を同じ長さのセグメントに分割する  $S_1, S_2, \dots, S_{\#Seg}$ 
5:   セグメント中の単語を特殊レベルが大きい方から小さい方への順で再配列する
6:   for  $i = 1$  to  $N/(BktSz * SegSz)$  do
7:     ActiveSeg =  $\phi$ 
8:     for  $j = 1$  to  $BktSz$  do
9:       ActiveSeg = ActiveSeg  $\cup S_{(j-1)N/(BktSz*SegSz)}$ 
10:    for  $j = 1$  to  $SegSz$  do
11:      新たなバケット  $B = \phi$  を作る
12:      ActiveSeg 中の全てのセグメントの  $j$  番目の単語を  $B$  に入れる
13:       $B$  を出力する

```

次ではアルゴリズム 2 で出力した単語列を単語バケットにする．アルゴリズム 2 がその過程を表している．バケットの大きさ $BktSz$ を $1 \leq BktSz \leq N$ に設定する．バケットの数が $\#Bkts = N/BktSz$ である．同じバケット中の単語を可能な限りに違う意味にするために単語列の $1, \#BktSz+1, 2 * \#BktSz+1, \dots, (BktSz-1) * \#BktSz+1$ 番目の単語をバケット 1 に， $2, \#BktSz+2, 2 * \#BktSz+2, \dots, (BktSz-1) * \#BktSz+2$ をバケット 2 に， $i, \#BktSz+i, 2 * \#BktSz+i, \dots, (BktSz-1) * \#BktSz+i$ をバケット i に入れる．図 3.1 が $N = 1000, BktSz = 2$ のときのバケット作り過程を表している．その操作により，2 つのバケット i と j の同じ位置の単語間の距離が同じ $\|i - j\|$ であり，意味的な距離の差も小さいと考えられる．またバケット同じ位置の単語間の距離が違う位置の単語間の距離より近いため，真の質問の単語が同じ位置にあると仮定する．したがって，真の質問の単語が意味的に近いときあるいは一つのトピックに集中したとき，バケットの中のダミー単語も同じように一つのトピックに集中すると考えられる．しかし，バケット中の単語の特殊レベルがランダムであり，大きく違う可能性がある．

図 3.2. バケット作り- $N = 1000, BktSz = 2, SegSz = 4$

バケット中の単語の特殊レベルを調整するために、隣接のバケット間の単語交換を行う．実際には単語を単語バケットに配置する前に単語列を同じ長さ $SegSz \leq N/BktSz$ のセグメントに分割し、セグメント内の単語を特殊レベルが大きい方から小さい方への順で再配列する． $SegSz$ が $BktSz$ の整数倍である必要がある．図 3.2 が図 3.1 の上に単語列の再配列を加えた流れを表している．その結果、同じバケットにある単語のセグメント内の順番が同一であり、特殊レベルが近くなると考えられる．

バケット作りには 2 つのパラメータを設定する必要がある． $SegSz$ が 2 つのリスクのトレードオフとなる． $SegSz$ が増加することは単語交換を行う範囲が増大することに相当する． $SegSz$ が大きければ大きいほどバケット中の単語の特殊レベルが近くなる．一方、単語間の意味的な距離も近くなる可能性がある．もう一つのパラメータ $BktSz$ がプライバシーと計算時間のトレードオフとなる． $BktSz$ が大きくなると、真の質問を特定する可能性が下がるが、検索サーバが処理する質問単語が増加する．

3.2.4 プライベート検索スキーム

本節では真の質問単語だけの関連値を計算できる検索スキームを述べる．検索スキームは質問加工、質問検索と結果処理 3 つの部分からなる．

アルゴリズム 4 が質問加工の流れを表す．真の質問単語が属するバケットの中の他の単語を全てダミー単語として質問に加える．ダミーを加えた質問の単語 t_j に $E(\mu_j)$ を付け、 t_j が真の質問単語なら $\mu_j = 1$ 、ダミー単語なら $\mu_j = 0$ ． $E(\cdot)$ は加算可能な準同型暗号 [Benaloh, 1994] の暗号化関数である．加算可能な準同型暗号が以下 2 つの特徴を持つ．二つの暗号文 $E(m_1), E(m_2)$ が与えられたときに、平文や秘密鍵なしで $E(m_1 + m_2)$ を計算できる．また、同じメッセージ m が複数の暗号文に対応でき、攻撃者が暗号文の頻度から m を推定することを防げる．

アルゴリズム 5 がサーバ側の検索過程を表す．サーバが単語と文書の関連値を保存している転置ファイルを用いて文書の関連値を計算する．加算可能な準同型暗号の特徴より、 $E(\mu_j)^{p_{ij}} = E(\mu_j * p_{ij})$ ． t_j がダミー単語であれば、 $E(score_j) * E(\mu_j)^{p_{ij}} = E(score_j) * E(0 * p_{ij}) = E(score_j)$ ．復号した関連値には影響を与えない．したがって、 $score_j$ が真の質問単語

Algorithm 4 質問加工1: **function** GENERATEBUCKETS(sq,BktSz,Segsz)**Input:** 真の質問単語 t_i の集合**Output:** 加工した質問 q

```

2:   for all 真の質問単語  $t_i$  do
3:     Bkt =  $t_i$  が属する単語バケット
4:     for all  $t_j \in \text{Bkt}$  do
5:       if  $t_i == t_j$  then  $\mu_j = 1$ 
6:       else  $\mu_j = 0$ 
7:        $E(u_j) = g^{\mu_j} \mu^r$ 
8:        $\langle t_j, E(\mu_j) \rangle$  を  $q$  に入れる

```

Algorithm 5 質問検索1: **function** GENERATEBUCKETS(sq,BktSz,Segsz)**Input:** 加工した質問 q **Output:** 文書とその文書暗号文した関連値の集合 R

```

2:    $R = \phi$ 
3:   for all  $\langle t_i, E(\mu_i) \rangle \in q$  do
4:     for all  $\langle d_j, p_{ij} \rangle \in L_i$  do
5:       if  $\exists \langle d_j, E(\text{score}_j) \rangle \in R$  then
6:          $E(\text{score}_j) = E(\text{score}_j) * E(\mu_j)^{p_{ij}}$ 
7:       else
8:          $\langle t_j, E(\mu_j)^{p_{ij}} \rangle$  を  $R$  に入れる

```

と文書の関連値 p_{ij} の和となる。

最後に質問者がサーバがらもらった結果集合の関連値を質問者だけが持っている秘密鍵で復号し、その値を用いて文書を再配列するとプライバシー保護手法を使っていない検索サーバと同様な検索結果がもらえる。

3.3 質問意図を曖昧化するキーワード検索

HDGA は [Pang et al., 2012] 提案した潜在ディリクレ配置法 (LDA) [Blei et al., 2003] に基づく質問意図の曖昧化メカニズム (TIO) の改良手法である。LDA の詳細は第 4 章で述べる。HDGA が以下の特徴を持つ、まず、サーバに提出した質問グループに属する各質問が違うトピックに属し、ダミー質問の生成過程が相互独立である。

次に、HDGA は TIO のように真の質問をカバーできるトピックからダミー質問を作るのではなく同じ質問グループに属する質問が同じ地位を持つようにする。

そして、HDGA がハッシュ関数 Highest Random Weigh(HRW) [Thaler and Ravishankar,

1998] を用いてダミートピックを選び, トピックの出現頻度を均一にする.

Algorithm 6 HDGA(On Masking Topical Intent in Keyword Search)

Input: 質問: q_1

- 1: $Q = \{q_1\} \delta_{q_1} = \underset{t \in T}{\operatorname{argmax}} Pr[t|q_1]$
- 2: **for all** $t \in T \setminus \{\delta_{q_1}\}$ **do**
- 3: $e_t = h(\delta_{q_1} || t || s)$
- 4: $T_D = \{t_{q_1}^1, t_{q_1}^2, \dots, t_{q_1}^2 | \forall t_1 \in T_D, \forall t_2 \in T \setminus T_D, e_{t_1} > e_{t_2}\}$
- 5: **for all** $t \in T_D$ **do**
- 6: **while** $\underset{t \in T}{\operatorname{argmax}} Pr[t|q'] \neq t$ **do**
- 7: $Pr[w|t]$ に基づいて $|q_1|$ 個の単語をランダムに選び, ダミー質問 q' を作る
- 8: $Q = Q \cup \{q'\}$
- 9: Q をシャッフルする

Output: Q

アルゴリズム 6 が HDGA の質問生成メカニズムを表す. ここで $Pt[w|t]$ が LDA 分析の結果であり, h が HRW ハッシュ関数である.

第 4 章

意味分析

第 3 章で述べたように質問意図を隠せるダミー質問を作るために質問が持つ意味を計算機に理解させなければならない．まず，文書や質問などをベクトルで表す必要がある．自然言語研究で多く使用されるベクトル表現方法が bag-of-words である．bag-of-words とは単語を袋に入れるように，単語の出現順番などの情報を捨て，単語の出現頻度だけをベクトル要素とする表現方法である．次に質問が持つ意味を数字にすると，質問をトピックベクトルで表現できる．したがって，質問が持つ意味を数字にすることは質問を記述するために必要な次元数を減らすことであると考えられる．情報検索分野ではコーパス中の文書を記述するために必要な次元数を減らす研究を進めている．

tf-idf はが単語とコーパス中の文書の関連値を実数値で表せるため，文書数 $|D| \times$ 単語数 $|W|$ の行列 M でコーパスを記述できる．潜在意味分析 (LSA) は行列 M を特異値分解 (SVD) し低ランク近似し，より高い圧縮を得る．しかし，LSA モデルのパラメータ数がコーパス中の文書数の共に増加するためオーバーフィットの恐れがあり，コーパスに含まれていない文書を記述する方法が明らかにさせていない．そのような問題を解決するために確率モデルである潜在ディリクレ配置法 (LDA) が提案された．

本論文ではこの 3 つの意味分析手法を全て用い，評価実験をする．以下では上記意味分析手法を紹介する．

4.1 tf-idf

コーパス中に含まれている文書の集合を D とし，単語 w_i の文書 d_j における出現回数を $n_{i,j}$ とする．単語 w_i の文書 d_j における出現頻度を

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \quad (4.1)$$

単語 i の逆文書頻度を

$$idf_i = \log \frac{|D|}{|d|w_i \in d, d \in D|} \quad (4.2)$$

と定義し，

$$tfidf_{i,j} = tf_{i,j} \cdot idf_i \quad (4.3)$$

によって単語 t_i の文書 d_j における $tf-idf$ 値を計算する．

本論文では国際特許分類を用い特許データベースに属する文書を 632 個の分類にし，各分類をトピックとし，

$$tfidf_{i,k} = \sum_{d_j \in t_k} tf_{i,j} \cdot idf_i \quad (4.4)$$

によって単語 t_i のトピック t_k における $tf-idf$ 値を計算する．人手により分類されている国際特許分類をトピックにしたため，各トピックの意味が明らかである．

4.2 潜在意味分析

特異値分解 (SVD) を用いて単語をトピック空間にマップすることが潜在意味分析の基礎である．LSA ではトピック空間中の単語と文書の関係を用いて多義性と同義性の問題を解決する．つまり，綴りが違うが同じような意味を持つ単語はトピック空間での距離が近いようにできる．

単語 \times 文書行列 M の (i, j) 番目の要素は単語 w_i のが文書 d_j における $tf-idf$ 値である． M を特異値分解 $M = USV^T$ し， U, S, V の各列ベクトルを特異値が大きい順に K 個用いて G の低ランク近似 $G_K = U_K S_K V_K^T$ を得る．このように低ランク分解によって，単語とトピックの関係を分析できる． M_K の (i, j) 番目の要素は i 番目の単語と j 番目のトピックの関係を表す．その値が大きければ大きいほど単語とトピックの関係が強い．単語 w_i と対応する行列 M_K の行 ℓ_i を単語 w_i のトピックベクトルとし，

$$rscore_{LSA}(q, t_j) = \sum_{w_i \in q} \ell_i[j] \quad (4.5)$$

によって質問 q とトピック t_j の関連値を計算する．

本論文では単語 \times 文書行列 M の代わりに単語 \times 国際特許分類行列 M' を用いる．単語 \times 国際特許分類行列 M' の (i, j) 番目の要素は単語 w_i の国際特許分類 t_k における $tf-idf$ 値である．国際特許分類は特許データベースの大きさと関係なく一定であるため，文書数の増加によるオーバーフィットを防ぐ．

4.3 潜在ディリクレ配置法

潜在ディリクレ配置法 (LDA) は文書の確率生成モデルである．LDA では文書が複数の潜在的トピックからランダムに生成されると仮定し，トピックをそのトピックごとに単語の出現頻度で表す．

LDA ではコーパス D に含まれている長さが n_d である文書 d の生成過程を以下のように仮定する:

- コーパス D における各潜在的トピック t を生成する $\phi_t \sim Dir(\beta)$
- トピック分布ベクトル θ_d を生成する $\theta_d \sim Dir(\alpha)$
- 各単語 $w \in d$ に対して:

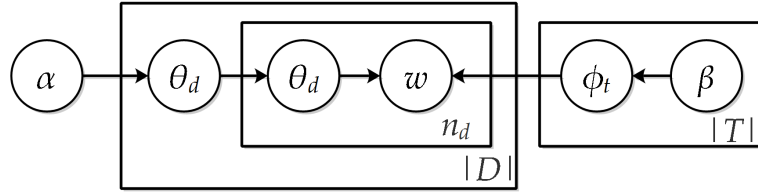


図 4.1. LDA のグラフィカルモデル

- w が属するトピック t を決める $t \sim \text{Multinomial}(\theta_d)$
- w を決める $w \sim \text{Multinomial}(\phi_t)$

ここで α, β は $|T|$ 次元ベクトルで、Dirichlet 分布のパラメータであり、 θ_d, ϕ_t は確率ベクトルである。

このグラフィカルモデルを図 4.1 に示す。

全ての文書が同じ確率を持つと仮定すると、

$$Pr[t] = \sum_{d \in D} \theta_{d,t} Pr[d] = \frac{\sum_{d \in D} \theta_{d,t}}{|D|} \quad (4.6)$$

によってコーパス D 中の各トピックの確率を計算し、

$$rscore_{LDA}(q, t) = Pr[t|q] = \frac{Pr[q|t]Pr[t]}{Pr[q]} = \frac{\prod_{w \in q} Pr[w|t]Pr[t]}{\sum_{t \in T} \prod_{w \in q} Pr[w|t]Pr[t]} \quad (4.7)$$

によって質問 q と各トピックの関係性を計算できる。

第 5 章

攻撃手法

本論文では攻撃者が質問者が質問意図を隠していることと質問者が用いている質問曖昧化手法のメカニズムを知っているを前提とし、攻撃手法を考える。

曖昧化検索は 3 つの違うレベルの目標があると同じように曖昧化検索に対する攻撃手法も 3 つの違うレベルの目標がある。1 つ目の攻撃手法ではダミー質問が混ぜられた質問グループから真の質問 q_R を見つける。2 つ目の攻撃手法ではダミー質問が混ぜられた質問グループから質問者が検索したいものを見つける。3 つ目の攻撃手法ではダミー質問が混ぜられた質問ログから質問者が興味を持つトピックを見つける。

1 つ目の目標に対して本論文では質問 q と質問 q のメイントピック δ_{SA} 間の関連値を攻撃するメイントピック攻撃を提案する。質問者が検索したいものを定義するのは難しいため、本論文ではダミー質問の検索結果と真の質問の検索結果が一致する割合を用いて評価する。そして、3 つ目の目標を達成できる既存の攻撃手法類似度攻撃 [Petit et al., 2016] を紹介し、改良手法を提案する。

5.1 メイントピック攻撃

HDGA など真の質問とダミー質問を 1 つの質問グループにして提出する曖昧化手法に対して、ダミー質問が真の質問と同様に全ての単語が 1 つのトピックに集中することが失敗したら、真の質問と真の質問のメイントピックの関連値がダミー質問とダミー質問のメイントピックの関連値より強いと考えられる。メイントピック攻撃 (MTA) は 1 つの質問グループの中で自分のメイントピックとの関連値が一番高い質問を真の質問とする。

また、ETSQ は真の質問単語ごとにダミー単語を混ぜ、1 つの加工した質問にして提出する。関係が強い単語が他のトピックと関係が強い単語の数より多いと加工した質問のトピックと真の質問のトピックが一致することが考えられる。また、一つのバケットの中の単語が意味的に遠いため、ダミー単語が真の質問単語のメイントピックとの関連値が低いと考えられる。メイントピック攻撃では各単語バケット中質問のメイントピックと一番関連値が高い単語を真の質問の単語と推定する。

5.2 事前情報がある場合の類似度攻撃

事前情報がある場合の類似度攻撃 (SimAtt)[Petit et al., 2016] は質問者が提出した質問と攻撃者が事前に得た質問者の質問ログ間の類似度を計算し、この類似度を用い質問者の新しい質問を見破る。SimAtt では単語ベクトルで質問を表し、質問ログを質問の集合とする。アルゴリズム 7 が質問 q と質問者のログ P_u 間の類似度 sim_{q,P_u} の計算方法を示す。

Algorithm 7 類似度計算

Input: 質問 q , 質問者のプロフィール P_u , スムージングパラメータ: α

- 1: **for** $q_i \in P_u$ **do**
- 2: $coef[i] \leftarrow 2 \cdot |q \cap q_i| \cdot \frac{1}{|q|+|q_i|}$
- 3: $coef \leftarrow sort(coef)$
- 4: $sim \leftarrow coef[0]$
- 5: **for** $i \in [1, |P_u|]$ **do**
- 6: $sim \leftarrow \alpha \cdot coef[i] + (1 - \alpha) \cdot sim$

Output: sim_{q,P_u}

ここで $coef[i]$ は質問 q と P_u に属する質問 q_i 間の Dice 係数 [Dice, 1945] で、 α が重み係数である。

Algorithm 8 SimAtt

Input: 質問グループ G , 質問者のプロフィール P_u , スムージングパラメータ: α

- 1: $q^* = \underset{q \in G}{\operatorname{argmax}} sim_{q,P_u}$

Output: q^*

SimAtt は質問グループの中質問者の質問ログと一番類似度が高い質問が真の質問であると判定する。攻撃方法は単純であるが、第 7 章の実験結果は特許データベースにおける攻撃の強さを示す。

5.3 事前情報がない場合の類似度攻撃

攻撃者が事前的に質問者の真の質問ログを持たないと SimAtt を用いることができない。本節ではダミー質問を混ぜた質問ログのみから真の質問を見つける事前情報がない場合の類似度攻撃 SimAtt2 を提案する。質問者が同じトピックに対して複数の質問を提出することが考えられる。PDS など真の質問と距離が大きい質問をダミー質問にする質問曖昧化手法では真の質問は同じトピック t に属するとしてもダミー質問は同じように別のトピック t' に属すると限らない。トピックの出現頻度から真の質問を見つけることができ、真の質問間の類似度がダミー質問間の類似度より高いと考えられる。HDGA はトピックの出現頻度を均一にできる。

しかし, HDGA は $Pr[w|t]$ に基づいて単語をランダムに選び, ダミー質問を生成するため, 同じトピックに属するダミー質問間の距離が真の質問間の距離ほど近いと限らない. また, 真の質問 q_1, q_2 が意味的に近いトピック t_1, t_2 に属するとき, ダミー質問 q'_1, q'_2 が属するトピック t'_1, t'_2 も意味的に近いと保証できない. したがって, 意味的に近い一連の質問が真の質問である可能性が高いと考えられる. 本論文では同じ質問者が提出した全ての質問グループから1つの質問を選び出した質問集合を質問列という. アルゴリズム9が意味的に近い質問列を取り出す攻撃手法を示す.

Algorithm 9 SimAtt2

Input: 質問グループ列 $\hat{G} = \{G_1, G_2, \dots, G_m\}$, スムージングパラメータ: α

```

1: for  $j \in |G_1|$  do
2:    $\hat{P}u[j] = G_1[j]$ 
3:    $\hat{P}ut[j] = \Phi$ 
4:    $d[j] = 0$ 
5: for  $i \in [2, m]$  do
6:   for  $j \in |G_i|$  do
7:      $\hat{P}ut[j] = \operatorname{argmax}_{Pu \in \hat{P}ut} \operatorname{sim}_{G_i[j], \hat{P}ut[j]}$ 
8:      $q_i^* = \operatorname{argmin}_{G_i[j] \in G_i} \operatorname{sim}_{G_i[j], \hat{P}ut[j]}$ 
9:     for  $j \in |Q_i|$  do
10:       $\hat{P}u[j] = \hat{P}ut[j] \cap G_i[j]$ 

```

Output: q^*

SimAtt2 は1つの質問グループに属する質問と同じ数の質問列 Pu を可能な真の質問列として保存する. 1つの質問グループに対してダミー質問を真の質問であると判定したとしても, その質問グループの中には必ず真の質問が存在するため, 真の質問と真の質問と一番意味的に近い質問列が保存し, 後続の質問グループに対する攻撃に利用される. したがって, SimAtt2 について過去の攻撃ミスは後続の攻撃に影響しないと考えられる.

第 6 章

単語ベクトルを用いた質問曖昧化

6.1 単語ベクトル

質問のメイントピック以外に質問に含まれている単語に情報も用いるため，本論文では単語ベクトルという概念を用いる．

定義 8. 単語ベクトル． W を全ての単語の集合とする．トピック t の単語ベクトル $wvec_{SA}(t)$ とは W に属する全ての単語 w を w と t の関連値を大きい方から小さい方まで並ぶ $|W|$ 次元のベクトルである．

$$wvec_{SA}(t) = (w_1, w_2, \dots, w_{|W|}) \quad (6.1)$$

ここで次の (i)，(ii) が成り立つ

- (i) $\forall w \in wvec_{SA}(t), w \in W$
- (ii) $\forall 1 \leq i < j \leq |W|, w_i \neq w_j, rscore(w_i, t) \geq rscore(w_j, t)$

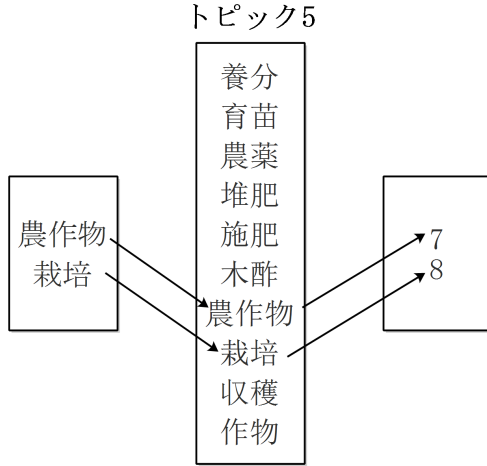
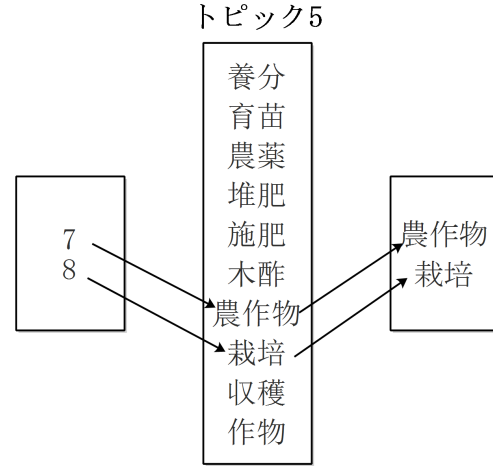
質問のメイントピックを計算し，質問に含まれている単語をその単語が質問のメイントピックの単語ベクトル内での順番にすれば，質問を数字ベクトルで表わすことができる．同様にトピックが決めれば，そのトピックの単語ベクトルを用いて数字ベクトルを質問に翻訳することができる．すなわち， $index(w, vec)$ をベクトル vec に元 w の順番を返す関数とし，質問数字化関数 WtN を

$$WtN_{SA}(q, t) = \{index(w, wvec_{SA}(t)) | w \in q\} \quad (6.2)$$

によって定義する．また $atindex(n, vec)$ をベクトル vec に順番が n となる元を返す関数とし，数字ベクトル質問化関数 NtW を

$$NtW_{SA}(v, t) = \{atindex(n, wvec_{SA}(t)) | n \in v\} \quad (6.3)$$

によって定義する．図 6.1，6.1 は LDA を意味分析ツールとして用いるときの例を表している．

図 6.1. 質問デジタル化関数 WtN の例図 6.2. 数字ベクトル質問化関数 NtW の例

6.2 質問者が検索したいトピックを曖昧化する質問曖昧化

検索したいトピックを曖昧化する質問曖昧化 (QOT) は以下の特徴を持つ。まず、トピック出現頻度で質問者が興味あるトピックを特定することを防ぐために QOT は事前にトピックをグループにし、質問者が検索したいトピックが属するトピックグループにある他のトピックをダミートピックとする。次に真の質問が同じ単語を含むとき、ダミー質問も同様に同じ単語を含むようにする。

それを実現するために提案手法では単語ベクトルを用いた。単語ベクトルに同じ順番を持つ単語がその単語ベクトルを持つトピックに対して同じ様な関連値を持つと考えられ、同じ数字ベクトルで表わせる質問もその質問が属するトピックに対して同じ様な関連値を持つと考えられる。したがって、単語ベクトルを通じて違うトピックに属するが似たような特徴を持つ質問を作ることができ、メイントピック攻撃に対応できると考えられる。

TG をトピックをグループにする関数とし、次のことが成り立つとする。

- (i) $\forall t \in T, TG(t) \subset T$
- (ii) $\forall t' \in TG(t), TG(t') = TG(t)$

QOT の質問生成メカニズムが次とおりである：

- (i) 質問 q のメイントピック $t_1 = \delta_{SA}(q)$ を計算する。
- (ii) トピック t_1 が属するトピックグループ $TG(t_1)$ を定める。
- (iii) 質問 q を数字ベクトル $v = WtN_{SA}(q, t_1)$ にする。
- (iv) 質問グループを $G = \{NtW(v, t) | t \in TG(t_1)\}$ とし、サーバに提出する。

質問者がトピック t_1 に対して一連の質問を提出するを例として QOT の安全性を分析す

る．トピック t_1, t_2, \dots, t_n が 1 つのトピックグループに属すると仮定する．質問者がメイントピックが t_1 である質問 q^1, q^2, \dots, q^m を QOT を用いて質問グループ G_1, G_2, \dots, G_m に曖昧化する．メイントピックが一致する質問ペア間の Dice 係数がメイントピックが一致ではない質問ペア間の Dice 係数より大きいと SimAtt2 で保存した m 個の質問列が t_1, t_2, \dots, t_n と 1 対 1 に対応することが考えられる．トピック t_i と対応する質問列を $Pu[i]$ ，質問グループ G_j の中の t_i に属する質問を q_i^j とする．質問グループ G_k に対して SimAtt2 攻撃するとき，質問ログに存在する任意の質問グループ G_i に対して $\forall 0 < i < k, coef(q^i, q^k) = coef(q_2^i, q_2^k) = \dots = coef(q_n^i, q_n^k)$ となり Dice 係数が同じである質問ペアが存在し， $sim_{q^k, Pu[1]} = sim_{q_2^k, Pu[2]} = \dots = sim_{q_m^k, Pu[m]}$ となるため，質問列との類似度も同じであり，SimAtt2 での攻撃が無効になる．

しかし，質問者が 1 つのトピックに対して検索すると限りない．特に特許検索の場合はいくつかの関係性が強い分野について検索することが多い．例えばスマートフォンメーカー企業がスマートフォン通信 (H セクション電気) の特許を検索した後にスマートフォン本体の生産装置 (B セクション処理操作) の特許を検索することもある．分野は違うが，同じスマートフォンに関する特許であるため，2 つの質問間の関連値が高いと考えられる．HDGA ではハッシュ関数でトピックをグループにしているが，各トピック間の関係を配慮していない．1 つのトピックグループ内のトピック間の類似度が QOT の SimAtt2 に対する安全性に影響すると考えられる．

トピックは単語との関連値で表し，一列に並べられないため，ETSQ が単語をバケットにすると同じようにトピックをグループにするができない．本論文では各トピックとの関連値が上位 1000 個までの単語をそのトピックを代表する単語集合とし，トピックを 1000 個の単語からなる質問とし，Dice 係数を用いてトピック間の距離を計算し，PDS で紹介した凝集型クラスタリングを用いてトピックをグループにする．7 章ではトピックグループ内のトピック間の類似度が安全性への影響を評価する．

6.3 質問者が検索したいトピックにおける質問曖昧化

攻撃が質問者の質問ログや質問者が興味を持つ分野など事前知識を持つとき，いくらダミートピックを増やしても，質問ログと類似度が高い質問や質問者が興味を持つ分野に属する質問などの方法で真の質問を簡単に見つかる．

質問者が検索したいトピックにおける質問曖昧化 (QOI) は単語ベクトルを用いて真の質問を数字ベクトルにし，数字ベクトルの各要素に対して雑音を加え，ダミー質問にする．

6.4 データベース分割

IPC コードを用いることにより特許データベースを分野ごとに子データベースに分割することができる．分割したデータベース各々に対して同じような信憑性を持つ質問を提出すると真に検索したいデータベースを隠すことができると考えられる．しかし，LSA や LDA など意

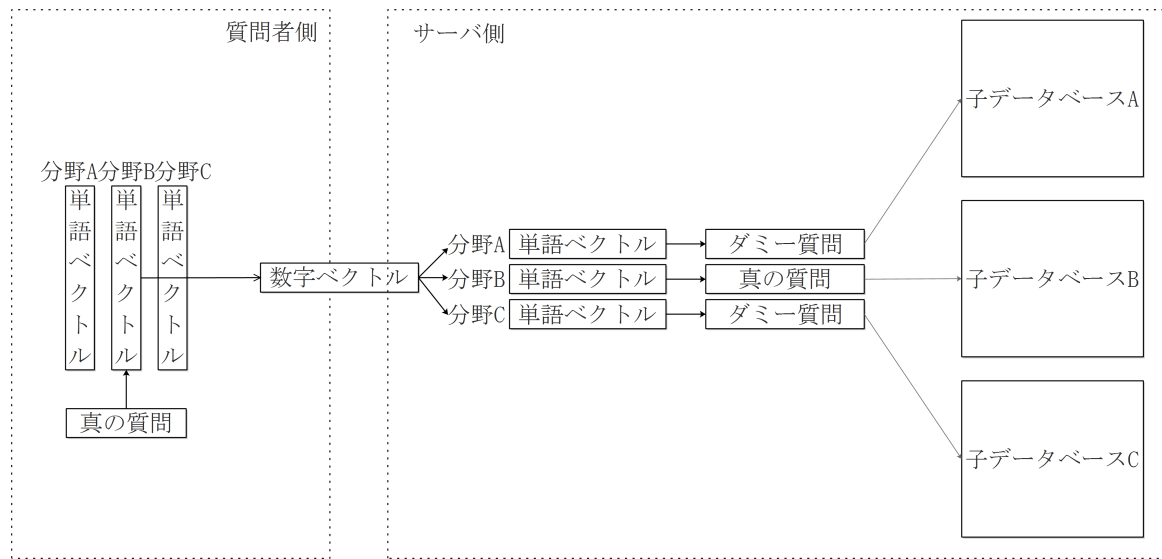


図 6.3. データベース分割の流れ

味分析手法を用いて得たトピックは人の手によって分類された子データベースと 1 対 1 に対応できない。本論文では検索サーバが第 4 章で紹介した *tfidf* を用いて単語と各子データベースの関連値を計算し、子データベースごとに単語ベクトルを作り、質問者に公開する。

質問者が検索するときは、真の質問を検索したい子データベースの単語ベクトルを用いて数字ベクトルにし、サーバに提出する。そして、サーバは質問者からもらった数字ベクトルを各子データベースに送り、各子データベースの単語ベクトルを用いて質問にし、各子データベースを検索する。図 6.3 はこの流れを表している。質問者は自ら意味分析を用いてダミー質問を作る必要がない。また、検索サーバでは各子データベースを並列で検索することができるため、ダミー質問による応答時間の増加を削減する。最後、質問者はサーバからもらった各子データベースの検索結果から真の子データベースの検索結果を利用する。

データベース分割では他の曖昧化と違って全ての子データベース、あるいはトピックについて質問を提出する。そのため、質問者が H セクションの特許について質問 q_1 を検索した後に B セクション特許について質問 q_2 を検索するとき、H セクションに属するダミー質問 q'_2 が必ず同じ質問グループに存在し、B セクションに属する真の質問 q_2 より前に提出した H セクションに属する質問 q_1 との類似度が高いと考えられる。SimAtt2 では意味的に近い一連の質問を真の質問とするため、真の質問 q_1 と q_2 ではなくより意味的に近い質問ペア q_1 と q'_2 を 1 つの質問列にする。そのため、データベース分割手法は SimAtt2 に対してよりいい防御ができると考えられる。

第 7 章

評価実験

本章では NTCIR-6[Fujii et al., 2007] の無効資料調査タスクのデータセットを用いて特許検索における既存手法 ETSQ, HDGA と提案手法を安全性を評価する．評価実験は全て 8 個の 2.5GHz CPU と 61GB メモリをもつ AWS Linux インスタンスで実行する．

7.1 文書集合と質問集合

NTCIR-6 は国立情報学研究所 (NII) から配布されている 1993 年から 2002 年まで発行分の約 350 万文書がある日本公開公報を検索対象である文書集合とする．無効資料調査タスクの質問は特許庁の審査感が拒絶した特許文書の一般的に最も重要な第一請求項を用いる．無効資料調査タスクの参加者は請求項を解析し単語を重要度を計算する手法や検索された文書の一部を用いて検索質問拡張を行う手法などを用いて検索精度を上げる．しかし，本論文は検索結果の精度について評価していないため，単純に請求項から名詞を抽出し検索質問をする．また第 4 章で説明したように本論文は IPC コードを用いて特許文書を 623 個の IPC サブクラスに分類する．文書集合と質問集合の詳細は表 7.1 に示す．

重複を除いた単語数	2,973,096
文書数	3,496,253
質問数	2,908
質問平均単語数	21.0
国際特許分類数	623

表 7.1. データセット

意味分析手法について，LSA と LDA は共に 64 トピックに設定する．特許文集集合の単語数が多く，全ての単語に対して LDA を行うことは困難であるため，本論文では各 IPC サブクラスとの *tf-idf* 値が上位 1000 個にある単語 32524 個に対して LDA を行う．評価実験で用いる質問の単語の 99.0% が上記 32524 個単語の中である．

本章で用いる略称は表 7.2 に表している．

7.2 メイントピック攻撃

7.2.1 質問者のプライバシーを保護する質問加工法

ETQS では任意 2 つの単語バケットで全ての同じ位置の単語ペア間の意味的距離の差が小さいなら、真の質問の単語が意味的に近いときあるいは一つのトピックに集中したとき、バケットの中のダミー単語も同じように一つのトピックに集中すると考えるため、本論文は SegSz を一番単語ペア間の意味的距離の差を小さくでき、攻撃しづらいと考えた 1 に設定する。また Wordnet は人の手によって作成されたため、27% の質問単語が Wordnet に存在しない、評価実験では Wordnet に存在しない単語を抜いて攻撃する。LSA と LDA を用いてメイントピック攻撃した結果が図 7.1 で表している。

単語ごとに 7 個のダミー単語を加えても 60% 以上の確率で真の質問の単語を見破られる。図 7.2 が加工した質問のメイントピックと真の質問のメイントピックが一致する確率を表している。ダミー質問の単語を 1 つのトピックに集中しないと真の質問のメイントピックを隠すことが困難であると考えられる。

7.2.2 質問意図を曖昧化するキーワード検索

HDGA はダミートピックの決定し、ダミートピックにおける単語の出現率でランダムに単語を選び真の質問と同じ長さのダミー質問を作る。攻撃者が質問者と同様に LDA を用いてメイントピック攻撃する結果が図 7.3 に示す。ここで \max は質問グループの中で自分のメイントピックの関連値 $Pr[\delta_{LDA}(q)|q]$ が一番高い質問 q が真の質問である確率で \min は質問グループの中で自分のメイントピックの関連値 $Pr[\delta_{LDA}(q)|q]$ が一番低い質問 q が真の質問である確率である。

HDGA ではダミートピック t を決定し、ダミートピックから $Pr[w|t]$ に基づいてランダム

手法	略称	節
潜在意味分析	LSA	4.2 節
潜在ディリクレ配置法	LDA	4.3 節
質問者のプライバシーを保護する質問加工法	ETSQ	3.2 節
質問意図を曖昧化するキーワード検索	HDGA	3.3 節
質問者が検索したいトピックを曖昧化する質問曖昧化	QOT	6.2 節
質問者が検索したいトピックにおける質問曖昧化	QOI	6.3 節
メイントピック攻撃	MTA	5.1 節
事前情報がある場合の類似度攻撃	SimAtt	5.2 節
事前情報がない場合の類似度攻撃	SimAtt2	5.3 節

表 7.2. 略称

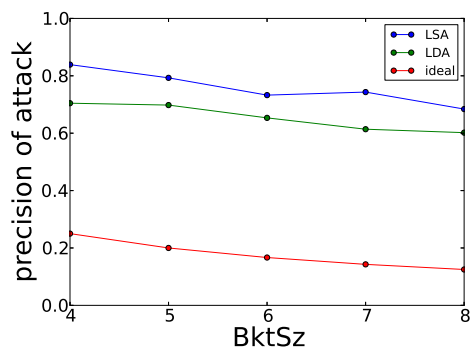


図 7.1. 単語パケットに対してメイントピック攻撃の成功率

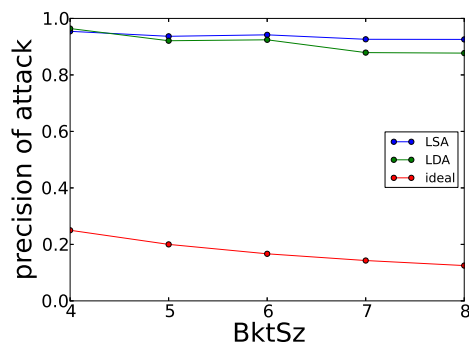


図 7.2. 加工した質問のメイントピックと真の質問のメイントピックが一致する確率

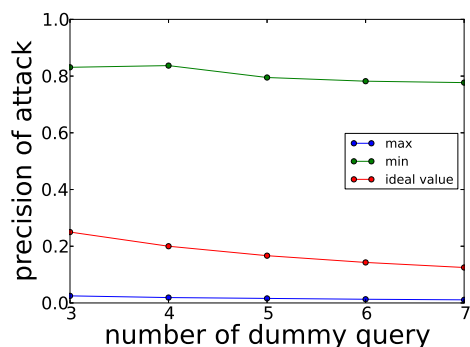


図 7.3. HDGA vs. MTA

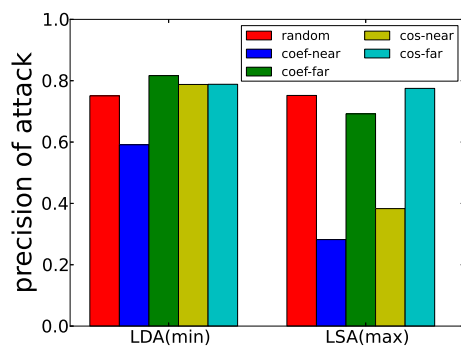


図 7.4. QOT vs. MTA

に単語を選択するため、各質問 q に対してトピック $\delta_{LDA}(q)$ における確率 $Pr[q|\delta_{LDA}(q)] = \prod_{w \in q} Pr[w|\delta_{LDA}(q)]$ の差が少ない。しかし、同じ文書に出現する確率が高い単語を用いた真の質問 q が出現する確率 $Pr[q]$ がダミー質問が出現する確率より大きくなるため、真の質問とその質問のメイントピックの関連値 $Pr[\delta_{LDA}(q)|q] = \frac{Pr[q|\delta_{LDA}(q)]Pr[\delta_{LDA}(q)]}{Pr[q]}$ が一番低くなると考えられる。本節では LDA を用いてメイントピック攻撃するときは関連値 $Pr[\delta_{LDA}(q)|q]$ が一番低い質問を真の質問とする。

7.2.3 質問者が検索したいトピックを曖昧化する質問曖昧化

攻撃者が質問者と同じ意味分析手法を用いて QOT に対してメイントピック攻撃を行うとき、1つのトピックグループ中のトピック間の距離の影響は図 7.4 に示す。ここで 1つの真の質問に対して 3つのダミー質問を加えるとする。第 4章で説明したように、coef 距離はトピックの代表単語集合間の Dice 係数で、cos 距離はトピックの代表単語集合を意味分析手法を用いて得たトピックベクトル間の cos 距離である。また、near は距離が小さいトピックを 1つのトピックグループにするとし、far は距離が大きいトピックを 1つのトピックグループにするとする。

LDA を用いるときは HDGA と同じように真の質問とその質問のメイントピックの関連値が一番低くなる確率が高い。意味的に近いトピックをダミートピックとにすることにより、トピックの出現率 $Pr[t]$ 間の差と単語ベクトルで同じ順番の単語 $|w|$ がそのトピックにおける出現率 $Pr[w|t]$ 間の差を小さくなり、HDGA によりいい結果を得ることができたと考えられるものの、真の質問が出現する確率とダミー質問が出現する確率間の差を無くすることができないため、理想値でメイントピック攻撃を防ぐことができないと考えられる。

LSA を用いて意味的に近いトピックと 1つのトピックグループにする場合は最も良い結果となった。意味的に近いトピックをダミートピックとすることにより、真の質問が属するトピック t とダミー質問が属するトピック t' の単語ベクトルに同じ順番 i を持つ単語 $w_i = wvec_{SA}(t)[i]$ と $w'_i = wvec_{SA}(t')[i]$ がそのトピックとの関連値 $rscore_{SA}(w_i, t)$ と $rscore_{SA}(w'_i, t')$ 間の差を無くすることが考えられ、LSA では質問が出現する確率を分析できないため、質問の出現率の差で真の質問を見つけることができないことが考えられる。

また、直接に単語を用いて距離を計算する場合が単語の曖昧性をなくす意味分析手法を用いる場合よりいい結果を得たのは特許文書は曖昧性を生じないように単語を選んだためと考えられる。

7.2.4 質問者が検索したいトピックにおける質問曖昧化

QOI に対しては QOT を攻撃するときと同じように攻撃者が質問者と同じ意味分析手法を用いて評価実験した結果は図 7.5 に示す。QOI は真の質問と同じトピックに属する似たような関連値を持つダミー質問を生成するが、LDA を用いる QOI は QOT で意味的に近いトピックを 1つのトピックグループにするときと同じような安全性を得る。同じトピックに属する質問をダミー質問にしても、真の質問の出現確率とダミー質問の出現確率間の差を無くすることができないと考えられる。

QOT の場合と同様に LSA を用いる QOI に対してメイントピック攻撃する場合は真の質問を見破られない。

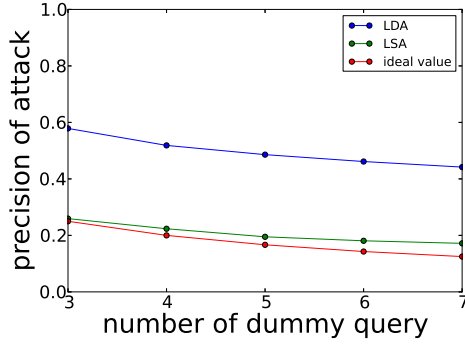


図 7.5. QOI vs. MTA

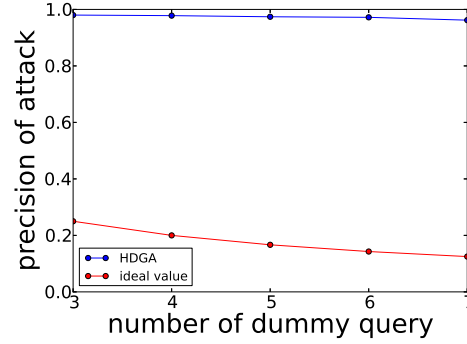


図 7.6. HDGA vs. SimAtt2

7.3 事前情報がない場合の類似攻撃

無効資料タスクで用いた質問は審査感が拒絶した特許文書から抽出されたものであるため，特許文書にあるほかの情報も利用できる．本論文では 1 つの企業が出願した特許文書から抽出された質問を 1 人の質問者が提出した質問にする．SimAtt2 の評価実験では 5 個以上の質問を提出した質問者 72 人が提出した 1562 個質問について攻撃する．

ETQS では 1 回の検索に対して 1 つの加工した質問しか提出しないため，類似攻撃は対応できない．

7.3.1 質問意図を曖昧化するキーワード検索

図 7.6 に示すように HDGA ではダミー質問の個数を 7 個まで増やしても 96.2% で真の質問を見つける．5.3 節で説明したように HDGA は質問が提出した一連の質問グループの質問間の関連値を配慮していないためと考えられる．

7.3.2 質問者が検索したいトピックを曖昧化する質問曖昧化

QOT に対して SimAtt2 を行うとき 1 つのトピックグループ中のトピック間の距離の影響は図 7.7 に示す．LSA を用いて意味的に近いトピックをダミートピックとすることにより，第 6.2 節で議論したように質問者が意味的に近いが同じトピックに属しない質問を連続提出した影響を減らし，HDGA に比べてよりいい結果を得ることができたが，3 つのダミー質問に対して 66.8% で真の質問を見つけることは安全であると言えない．

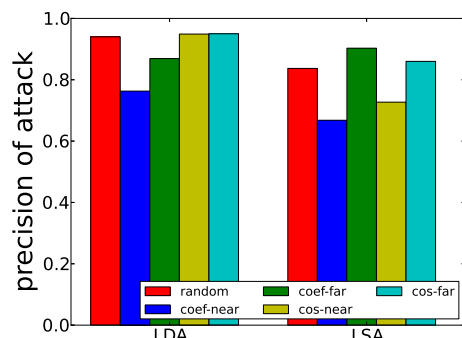


図 7.7. QOT vs. SimAtt2

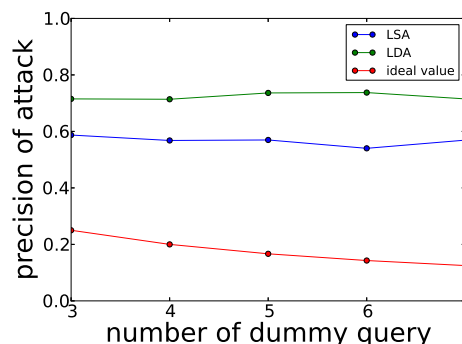


図 7.8. QOI vs. SimAtt2

7.3.3 質問者が検索したいトピックにおける質問曖昧化

QOI では 1 つの質問グループの中の質問全部 1 つのトピックに属し、真の質問が意味的に近いときダミー質問も意味的に近いと考えられる。しかし、特許文書では曖昧性を生じないように同じものを指すときは同じ単語を用いるため、同じ質問が提出した質問が意味的に近いだけでなく同じ単語を用いる確率も高い。そのため、単語の重複率で質問間の類似度を計算する類似度攻撃は一般的なウェブ検索に攻撃するときよりいい効果を得たと考えられる。その結果を図 7.8 に示す。

7.4 事前情報がある場合の類似攻撃

類似攻撃の評価実験では SimAtt2 で用いた各質問者の質問からランダムに 3 つを選び、その質問者が攻撃者に知られている質問ログとし、その質問ログを用いて類似攻撃する。

実験結果は図 7.9 に表している。QOT と QOI では SimAtt2 に対して一番いい結果を得た意味分析手法 LSA と coef 距離が近いトピックグループを用いた。

評価実験で用いた質問では質問の 22% の単語が攻撃者に知られている質問ログに存在するため、単純な質問曖昧化手法で類似攻撃から質問意図を守ることができないと考えられる。

7.5 データベース分割

本論文では国際特許分類の一番上の階層の 8 個のセクションを用いて特許文書を 8 個の子データベースに分割し、評価する。評価実験結果は図 7.10 に表している。

データベース分割は LDA を用いたメイントピック攻撃 MTA-LDA に対しては理想値に

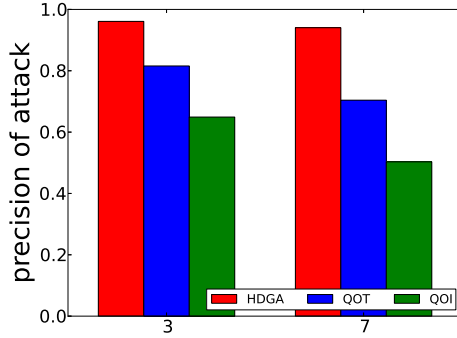


図 7.9. SimAtt

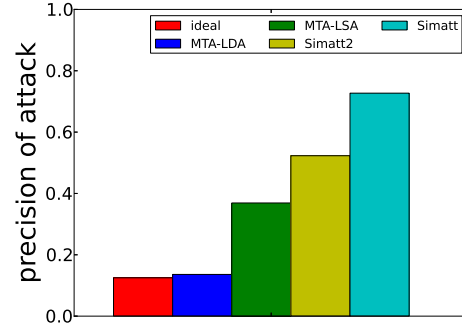


図 7.10. データベース分割

近い高い性能を出している．このことから，子データベースにおける tf-idf 値に基づいて作られたダミー質問 q' がその質問の LDA におけるメイントピック $\delta_{LDA}(q')$ における出現率 $Pr[q'|\delta_{LDA}(q')]$ が真の質問 q の出現率 $Pr[q|\delta_{LDA}(q)]$ より低いと考えられるものの，ダミー質問 q' の出現率 $Pr[q']$ も真の質問より低いいためメイントピックとの関連値 $Pr[\delta_{LDA}(q)|q]$ と $Pr[\delta_{LDA}(q')|q']$ が近くなったと考えられる．このことを確かめるため真の質問とダミー質問の出現率を調べた．36.49% の真の質問の出現率がダミー質問の出現率より高い結果となった．また，LSA を用いたメイントピック攻撃 MTA-LSA は 36.90% で真の質問を見つける．このことから，単語と子データベース間の tfidf 値が単語とトピックの関連値を表せると考えられるものの，子データベースの数が少ない，LDA と LSA を用いて得たトピックと対応できないため，LDA と LSA における多次元なトピック空間を完璧に記述することができないと考えられる．

類似攻撃に対しては全てのトピックに対して質問を提出することよりトピック間の差による質問間の類似度の差と質問と質問が属するトピックの関連値の差が小さくなるが，トピック数が減らしたため，同じトピックに属するダミー質問が意味的に遠くなり，攻撃されやすくなると考えられる．

7.6 交差攻撃

今までの評価実験では既存研究と同様に攻撃者と質問者が同じ意味分析手法を用いた．しかし，LDA では 1 つのトピック集中している単語は LSA を用いても同じく 1 つのトピック集中すると限らない．本節では攻撃者が質問者と違う意味分析手法を用いてメイントピック攻撃を行った．

図 7.11，7.12 は交差攻撃の結果を表している．LDA モデルのトピックと LSA モデルのトピックは一対一に対応できないため，両方とも攻撃者が質問者と同じ意味分析手法を用いた場

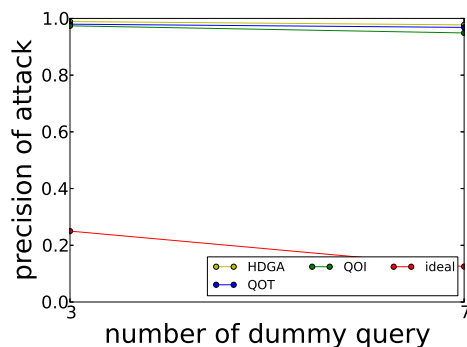


図 7.11. 質問者：LDA vs. 攻撃者：LSA

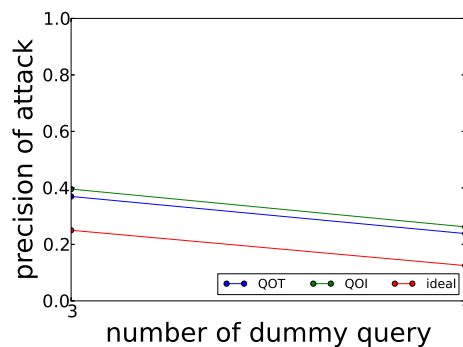


図 7.12. 質問者：LSA vs. 攻撃者：LDA

合より良い攻撃結果となった．ダミートピックからランダムに生成したダミー質問は真の質問のように複数の意味分析に対応できる強さを持たないと考えられる．また，LSA を用いた場合，LDA を用いた結果を大きく上回っている．このことから，単語数が多いが文章数が少ないコーパスに対して意味分析する場合は LSA が LDA より良い効果を出すと考えられる．

7.7 検索曖昧化手法の比較

全体的に各検索曖昧化手法を比較するために，各攻撃方法を用いて各検索曖昧化手法を攻撃した結果を 1 つの表にまとめた．1 つの真の質問に対して一緒に検索サーバに提出するダミー質問を 7 つに設定した．また，QOT に対しては coef 距離が近いトピックを 1 つのグループにした．この結果は表 7.3 に示す．

	攻撃者				
		MTA-LSA	MTA-LDA	SimAtt2	SimAtt
質問者	ETSQ	68.4	60.2	x	x
	HDGA	97.7	77.7	96.2	94.0
	QOT-LSA	12.2	23.8	49.4	70.4
	QOT-LDA	96.8	55.7	88.9	94.3
	QOI-LSA	17.2	26.2	56.9	50.3
	QOI-LDA	94.5	44.2	71.5	81.2
	データベース分割	36.9	13.6	52.3	72.7

表 7.3. 検索曖昧化手法の比較

事前情報を持たない攻撃者に対しては QOT-LSA が一番いい結果を得たが，事前情報を

持つ攻撃者に対しては QOI-LSA が一番いい結果を得た．QOT と QOI は相互影響しないため，両方同時に用いる手法の評価は今後の課題として挙げられる．

第 8 章

おわりに

本論文では特許検索における曖昧化検索手法と攻撃手法を提案し，既存の曖昧化検索手法に対して実データを用いて手法の安全性を評価した．評価実験結果により，提出しようとしている質問のみからダミー質問を生成する手法は質問ログを持つ攻撃者から質問意図を保護することが困難であると考えられる．また，どのような意味分析手法においても同じような強さを持つダミー質問を生成することが今後の課題として挙げられる．

謝辞

本論文は多くの方々のご指導ご支援の下，書き上げることができました．指導教員の中川先生には，多くのところをご教授してくださりました．テーマの設定のしかたや関連研究の調べ方，論文の書き方など研究活動に必要な多くのことを学びました．荒井助教授には研究テーマの選別，実験データの作成などで大変お世話になりました．

また，同研究室の後輩，同期，先輩方一同にも，発表等において毎回数多くのご指摘を頂きました．特に博士の小宮山さんは研究だけではなく，その他生活全般についてお世話になっていたものと思っています．

以上簡単ではありますが，まとめて感謝の意を記させていただきたいと思います．

参考文献

- [Balsa et al., 2012] Balsa, E., Troncoso, C., and Diaz, C. (2012). OB-PWS: Obfuscation-Based Private Web Search. In *2012 IEEE Symposium on Security and Privacy*, pages 491–505.
- [Benaloh, 1994] Benaloh, J. (1994). Dense probabilistic encryption. In *Proceedings of the workshop on selected areas of cryptography*, pages 120–128.
- [Bethencourt et al., 2006] Bethencourt, J., Song, D., and Waters, B. (2006). New constructions and practical applications for private stream searching. In *2006 IEEE Symposium on Security and Privacy (S&P'06)*, pages 6–pp. IEEE.
- [Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.
- [Bodon, 2010] Bodon, F. (2010). A fast apriori implementation. In *Proceedings of the IEEE ICDM workshop on frequent itemset mining implementations (FIMI '03)*, volume 90.
- [Boneh et al., 2004] Boneh, D., Di Crescenzo, G., Ostrovsky, R., and Persiano, G. (2004). Public key encryption with keyword search. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 506–522. Springer.
- [Chor et al., 1998] Chor, B., Kushilevitz, E., Goldreich, O., and Sudan, M. (1998). Private information retrieval. *Journal of the ACM (JACM)*, 45(6):965–981.
- [Deerwester et al., 1990] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41(6).
- [Dice, 1945] Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302.
- [Dingledine et al., 2004] Dingledine, R., Mathewson, N., and Syverson, P. (2004). Tor: The second-generation onion router. Technical report, DTIC Document.
- [Freedman et al., 2005] Freedman, M. J., Ishai, Y., Pinkas, B., and Reingold, O. (2005). Keyword search and oblivious pseudorandom functions. In *Theory of Cryptography Conference*, pages 303–324. Springer.
- [Fujii et al., 2007] Fujii, A., Iwayama, M., and Kando, N. (2007). Overview of the Patent

- Retrieval Task at the NTCIR-6 Workshop. In *NTCIR*.
- [Josep Domingo Ferrer et al., 2009] Josep Domingo Ferrer, Agusti Solanas, and Jordi Castell Roca (2009). $h(k)$ private information retrieval from privacy uncooperative queryable databases. *Online Information Review*, 33(4):720–744.
- [Michael and Tom, 2006] Michael, B. and Tom, Jeller, J. (2006). A Face Is Exposed for AOL Searcher No. 4417749 - New York Times.
- [Miller, 1995] Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.
- [Murugesan and Clifton, 2008] Murugesan, M. and Clifton, C. (2008). Plausibly deniable search. In *Proceedings of the Workshop on Secure Knowledge Management (SKM 2008)*.
- [Murugesan and Clifton, 2009] Murugesan, M. and Clifton, C. (2009). Providing Privacy through Plausibly Deniable Search. In *Proceedings of the 2009 SIAM International Conference on Data Mining*, Proceedings, pages 768–779. Society for Industrial and Applied Mathematics.
- [Pang et al., 2010] Pang, H., Ding, X., and Xiao, X. (2010). Embellishing Text Search Queries to Protect User Privacy. *Proc. VLDB Endow.*, 3(1-2):598–607.
- [Pang et al., 2012] Pang, H. H., Xiao, X., and Shen, J. (2012). Obfuscating the Topical Intention in Enterprise Text Search. In *2012 IEEE 28th International Conference on Data Engineering*, pages 1168–1179.
- [Petit et al., 2016] Petit, A., Cerqueus, T., Boutet, A., Mokhtar, S. B., Coquil, D., Brunie, L., and Kosch, H. (2016). SimAttack: private web search under fire. *Journal of Internet Services and Applications*, 7(1):1.
- [Saint-Jean et al., 2007] Saint-Jean, F., Johnson, A., Boneh, D., and Feigenbaum, J. (2007). Private web search. In *Proceedings of the 2007 ACM workshop on Privacy in electronic society*, pages 84–90. ACM.
- [Shapira et al., 2005] Shapira, B., Elovici, Y., Meshiach, A., and Kuflik, T. (2005). PRAW A PRivAcY model for the Web. *Journal of the American Society for Information Science and Technology*, 56(2):159–172.
- [Song et al., 2000] Song, D. X., Wagner, D., and Perrig, A. (2000). Practical techniques for searches on encrypted data. In *Security and Privacy, 2000. S&P 2000. Proceedings. 2000 IEEE Symposium on*, pages 44–55. IEEE.
- [Thaler and Ravishankar, 1998] Thaler, D. G. and Ravishankar, C. V. (1998). Using name-based mappings to increase hit rates. *IEEE/ACM Transactions on Networking (TON)*, 6(1):1–14.
- [Wang and Ravishankar, 2014] Wang, P. and Ravishankar, C. V. (2014). On masking topical intent in keyword search. In *2014 IEEE 30th International Conference on Data Engineering*, pages 256–267. IEEE.
- [Zobel and Moffat, 2006] Zobel, J. and Moffat, A. (2006). Inverted Files for Text Search

Engines. *ACM Comput. Surv.*, 38(2).