修士論文

言語モデルを用いた Q&A アーカイブにおける類似質問検索

(Similar Question Retrieval in Q&A Archives Based on Language Model)

高橋 輝

東京大学大学院 情報理工学系研究科 電子情報学専攻

指導教員 安達 淳

2010 年 2 月 9 日提出

概要

近年，一般ユーザが情報を発信する機会が増え続けている．そのような機会の中にコミュニティベースの Q&A (cQA) サイトが挙げられる．cQA はコミュニティのユーザが質問を投稿し，他のユーザが回答するサービスである．サービスの数やユーザが増していくに連れて，cQA サイトのアーカイブデータは豊富な情報源となっている．

この情報にアクセスするためには，Q&A アーカイブにおける検索が必要である．これはクエリ質問と類似した質問をアーカイブから検索することである．通常の文書検索と比較してのメリットとして，クエリが自然文なので情報要求をより的確に表現できること，関連する文書ではなく質問への回答を直接得られること，等がある．

類似質問検索には，文書が短いため word overlap が少ないという特有の難しさがある．これに対して取り組んだ既存手法では，複数の確率的モデルを手動で組み合わせたモデルを用いている．筆者はこの手法を拡張し，混合モデルとしての統一的なフレームワークを提案する．合わせて，混合モデルの構成要素として効果的なものの一例を提示する．

混合モデルでは混合比率の推定が必要となる．筆者は混合重みの扱いについて，Q&A によらず一定とするものと，Q&A ごとに異なるとする 2 つの方法を提案する．また，どちらの場合でも，訓練データからのモデルのサンプリング法およびその結果を用いた混合比率の推定方法を述べる．構成要素としては，既存手法において用いられたモデルをベースに考える．それらのモデルの表現する情報を考察し，質問文という言語現象を効果的に表現する構成要素を提案する．

これらの理論の有効性を検証するために実験を行う．まず，混合比率の推定について，効果的に推定するための学習データと基本モデルの組み合わせを模索する．次に，モデルに出現するパラメータの最適な値を，学習データサイズとの比較から検討する．最後に，上の 2 つの実験から得られた知見を基に，提案手法の性能を検証する．混合比率を一定とする手法と，Q&A ごとに異なるとする手法の性能を比較する．加えて，類似質問検索の既存手法とも性能を比較し，提案手法

の有効性が確認されたことを報告する.

目次

図目次

表目次

第 1 章
序論

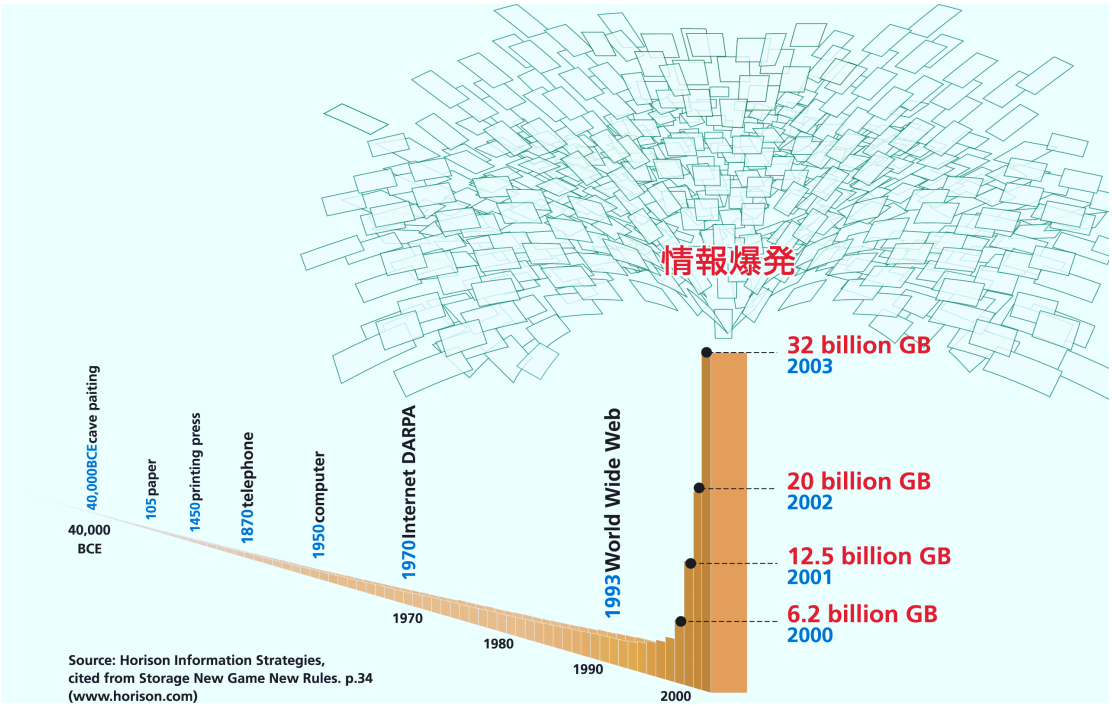

図 1.1: 情報爆発時代　出処：特定領域研究『情報爆発時代に向けた新しい IT 基盤技術の研究』

1.1　研究の背景と目的

インターネットの登場により，人類の生産する情報は爆発的に増加し続けている [BS09]．図 1.1 からもわかるように，年々その勢いを増している．その要因として，一般ユーザによる情報の発信が挙げられる．一般ユーザによる情報発信の例としてはブログや，情報交換のためのコミュニティが考えられる．その中で，新たな知識源としてコミュニティベースの Q&A サイト (cQA) に注目が集まっている．cQA はコミュニティのユーザが質問を投稿し，他のユーザが回答する Web サービスである．質問と回答の例を図 1.2 に示す．cQA サービスは国内外に多数存在し，一日に大量の質問が投稿・回答される．従って，そのアーカイブデータは非常に豊かな知識源と言える．

Q&A アーカイブはいわば，質問文すなわち形式化された情報要求 (formalized need)[Tay67] をキーとしてアクセス可能な知識データベースである．筆者は Q&A アーカイブからの情報検索の実現を目指している．Q&A 検索と通常の Web 検索が異なる点は大きく 2 つある．

1. クエリは単語でなく文章である

図 1.2: cQA サービスにおける質問と回答の例

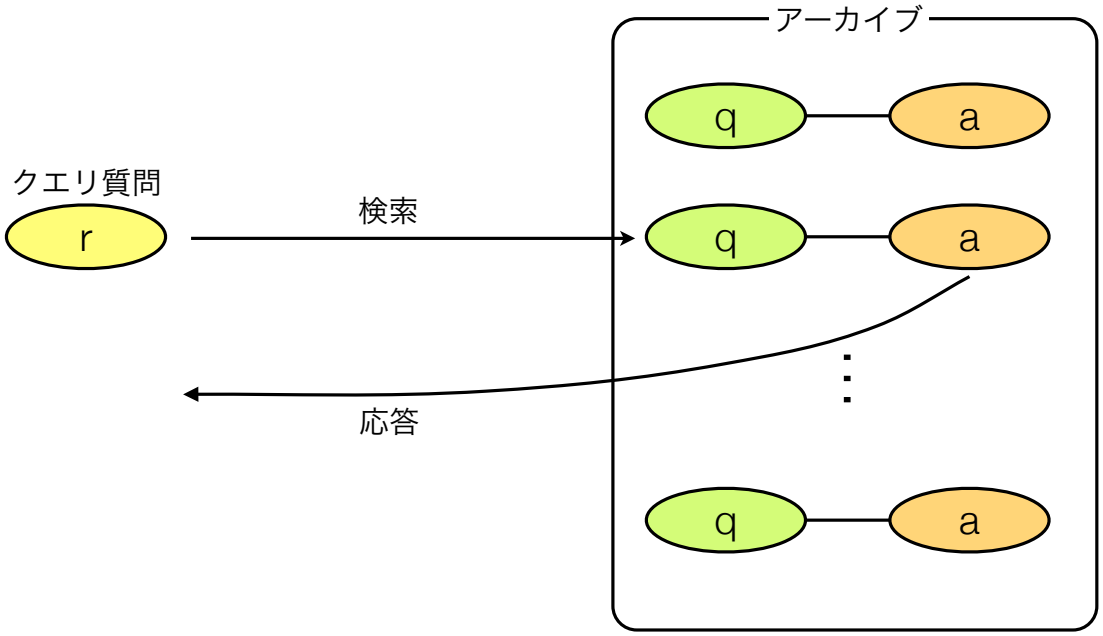

図 1.3: Q&A 検索モデル

これにより，ユーザの情報要求をより明確に表現することができる．

2. システムが返すのは適合文書ではなく質問への回答である
   これによりユーザは文書から回答を探す必要がない．

この 2 点を考えると，Q&A 検索は質問応答を検索により実現する手法と捉えることができる．すなわち，アーカイブ中の質問から，クエリ質問と類似するものを検索することにより，質問応答を実現することができる．こうして Q&A 検索はアーカイブから類似質問を検索することによって実現できる (図 1.3).

類似質問検索は，クエリとして与えられた質問文に類似した内容の質問をアーカイブから検索するタスクである．類似質問検索は通常の文書検索と異なり，対象文書のサイズが小さいため，質問に含まれる語が検索対象となる質問文に現れないことが多い．そこで，単語の意味の類似性を考慮する必要がある．また，検索対象の質問文だけでなく，それに対応する回答文情報を利用することで，高い精度を実現することが期待できる．これらのことから，複数の情報をうまく組み合わせた検索モデルを構築することが有効であると考えられる．

例えば文献 [XJC08] では，複数の確率的モデルを手動で組み合わせることで性能を向上させている．筆者はこの手法を拡張し，Q&A ペアを複数の情報による基本モデルを混合した言語モデルで表現することを提案する．合わせて，任意の基

本モデルを混合する際の混合比率の推定方法を提案する．また，質問文という言語現象をよく表現する基本モデルの組み合わせを提案する．

1.2 本論文の構成

本稿では，第 2 章にて関連研究について説明する．提案手法を実現するための要素技術と，cQA の情報源としての利用に関する研究を紹介する．第 3 章では類似質問検索の手法を提案する．第 4 章では実験について説明する．混合モデルの実現の可能性を探る実験と，性能を評価するための実験を行った．最後に第 5 章にて今後の課題と共にその発展の方向性を示唆する．

第 2 章
関連研究

2.1 ベイズ統計の基礎

本論文では確率に基づく検索手法を提案している．本節ではまずベイズの定理周辺の基礎について簡単に触れ，グラフィカルモデルや確率の推定手法について紹介する．

2.1.1 ベイズの定理

事象 A が起こったときに事象 B が起こる確率を，事象 A のもとでの事象 B の条件付き確率 (conditional probability) と呼び，$P(B|A)$ と表す．条件付き確率は次のように与えられる．

$$P(B|A) = \frac{P(A,B)}{P(A)} \tag{2.1}$$

ここで $P(A,B)$ は事象 A と B が同時に起こる確率を表し，結合確率 (joint probability) と呼ばれる．一方，条件付き確率 $P(A|B)$ は

$$P(A|B) = \frac{P(A,B)}{P(B)} \tag{2.2}$$

であるから，

$$P(A,B) = P(A|B)P(B) \tag{2.3}$$

となり，これを式 (2.1) に代入して

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)} \tag{2.4}$$

を得る．これをベイズの定理という．

ベイズの定理の適用例を挙げる．

サイコロやくじ引きのように，K 通りの異なる結果のいずれか 1 つが得られる試行を繰り返すことを考える．結果 1 を得る確率を θ_1，結果 2 を得る確率を θ_2，…，結果 K を得る確率を θ_K とする．このとき，結果 1 が x_1 回，結果 2 が x_2 回，…，結果 K が x_K 回起きる確率は

$$P(\boldsymbol{x}|\boldsymbol{\theta}) = \frac{(\sum_k x_k)!}{\prod_k x_k!} \prod_{k=1}^{K} \theta_k^{x_k} \tag{2.5}$$

となる．ここで $\boldsymbol{x} = (x_1, x_2, \ldots, x_K)$, $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$ である．このような式で表される確率分布を多項分布 $\mathcal{MULTI}(\boldsymbol{\theta})$ と呼ぶ．このようなデータ $\boldsymbol{x}$ が実際に得

られたとき，式 (2.5) はもはや確率ではなく，尤度 (likelihood) と呼ばれる．ここで，パラメータ $\boldsymbol{\theta}$ の扱いに関する問題を考える．

頻度主義の考えに従えば，尤度関数を最大にするようにして $\boldsymbol{\theta}$ の値を推定できる．尤度関数の最大化は，対数尤度の最大化と等価なので，多項分布の対数尤度関数を考える．

$$\ln P(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{k=1}^{K} x_k \ln \theta_k + \text{const.} \tag{2.6}$$

これを最大化するには，ラグランジュ乗数 λ を導入して，

$$\sum_{k=1}^{K} x_k \ln \theta_k + \lambda(\sum_{k=1}^{K} \theta_k - 1) \tag{2.7}$$

を最大化する．$\boldsymbol{\theta}$ で微分した導関数が $\mathbf{0}$ となることにより，最尤推定量

$$\theta_k = \frac{x_k}{\sum_i x_i} \tag{2.8}$$

を得る．

よりベイズ的なアプローチのために，ベイズの定理を用いる．データ $\boldsymbol{x}$ が得られたときの $\boldsymbol{\theta}$ の確率分布 $P(\boldsymbol{\theta}|\boldsymbol{x})$ は，ベイズの定理により次のように展開される．

$$P(\boldsymbol{\theta}|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|\boldsymbol{\theta})P(\boldsymbol{\theta})}{P(\boldsymbol{x})} \tag{2.9}$$

分母 $P(\boldsymbol{x})$ は $\boldsymbol{\theta}$ によらないので

$$P(\boldsymbol{\theta}|\boldsymbol{x}) \propto P(\boldsymbol{x}|\boldsymbol{\theta})P(\boldsymbol{\theta}) \tag{2.10}$$

となる．ここで $P(\boldsymbol{\theta})$ は $\boldsymbol{x}$ が得られる前の $\boldsymbol{\theta}$ の確率分布であり，事前分布 (prior) と呼ばれる．それに対し，$P(\boldsymbol{\theta}|\boldsymbol{x})$ は事後分布 (posterior distribution) と呼ばれる．同じ $\boldsymbol{\theta}$ の確率分布でも，$\boldsymbol{x}$ を得る前と得た後では異なる．ここで，式 (2.5) の尤度関数 $P(\boldsymbol{x}|\boldsymbol{\theta})$ は $\theta_k^{x_k}$ の形の因数の積になっている．従って，もし事前分布を $\theta_k^{x_k}$ の積に比例するように選ぶと，事後分布は，事前分布と尤度関数の積に比例するので，事前分布と同じ関数形式になる．そのような事前分布を共役事前分布 (conjugate prior) と呼ぶ．多項分布の共役事前分布はディリクレ分布 (Dirichlet distribution) と呼ば

れ，確率密度関数は次の式で表される．

$$P(\boldsymbol{\theta};\boldsymbol{\alpha}) = \mathcal{DIR}(\boldsymbol{\alpha}) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \tag{2.11}$$

ここで Γ はガンマ関数で,

$$\Gamma(x) \equiv \int_0^\infty u^{x-1} e^{-u} du \tag{2.12}$$

で定義される．また式 (2.11) 中の係数は，ディリクレ分布が正規化されることを保証している．すなわち

$$\int \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} d\boldsymbol{\theta} = \frac{\prod_k \Gamma(\alpha_k)}{\Gamma(\sum_k \alpha_k)} \tag{2.13}$$

が成り立つ．ディリクレ分布の平均 (mean) は

$$\mathbb{E}[\boldsymbol{\theta}] = \frac{\boldsymbol{\alpha}}{\sum_i \alpha_i} \tag{2.14}$$

で与えられる．一方，最頻値 (mode) は

$$\frac{\boldsymbol{\alpha} - \mathbf{1}}{\sum_i \alpha_i - K} \tag{2.15}$$

である．

ディリクレ事前分布のパラメータ $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_K)$ は確率分布 $P(\boldsymbol{x}|\boldsymbol{\theta})$ のパラメータ $\boldsymbol{\theta}$ の確率分布のパラメータであるから，超パラメータ (hyper parameter) と呼ばれる．

式 (2.10) と式 (2.5)，式 (2.11) から，事後分布は

$$P(\boldsymbol{\theta}|\boldsymbol{x};\boldsymbol{\alpha}) \propto \prod_{k=1}^{K} \theta_k^{x_k + \alpha_k - 1} \tag{2.16}$$

となり，正規化係数は式 (2.11) との比較から

$$P(\boldsymbol{\theta}|\boldsymbol{x};\boldsymbol{\alpha}) = \frac{\Gamma \sum_k (x_k + \alpha_k)}{\prod_k \Gamma(x_k + \alpha_k)} \prod_{k=1}^{K} \theta_i^{x_k + \alpha_k - 1} \tag{2.17}$$

となる．この式から，データ集合 $\boldsymbol{x}$ を観測したあとの事後分布を求めるには，すべての k について α_k の値を x_k だけ増やせばよいことがわかる．このことから，超パラメータは各結果の有効観測数 (effective number of observations) として，簡潔

に解釈できる．ただし，$\boldsymbol{\alpha}$ の各成分は整数である必要はない．

事前分布を導入した上で，パラメータ $\boldsymbol{\theta}$ を推定するには，最大事後確率 (maximum a posteriori: MAP) 推定を行う．これは事後確率が最大になるような変数の値をもって推定値とするものである．すなわち最頻値のことで，ディリクレ分布の最頻値は式 (2.15) であるから， MAP 推定量は

$$\theta_k = \frac{x_k + \alpha_k - 1}{\sum_i (x_i + \alpha_i) - K} \tag{2.18}$$

となる．

ここまでの議論は，頻度主義の考えに基づくパラメータの推定である．完全なベイズ主義では，$\boldsymbol{\theta}$ は観測されていないので，確率的に変動する量として捉えられる．従って，パラメータ $\boldsymbol{\theta}$ は決定できない．しかし，次に同じ試行を行った時の結果を予測することはできる．これは予測分布 (predictive distribution) と呼ばれる．次の試行により結果 k を得る確率は

$$\begin{aligned} P(k|\boldsymbol{x};\boldsymbol{\alpha}) &= \int P(k,\boldsymbol{\theta}|\boldsymbol{x};\boldsymbol{\alpha}) d\boldsymbol{\theta} \\ &= \int P(k|\boldsymbol{\theta},\boldsymbol{x};\boldsymbol{\alpha}) P(\boldsymbol{\theta}|\boldsymbol{x};\boldsymbol{\alpha}) d\boldsymbol{\theta} \end{aligned} \tag{2.19}$$

次の試行はデータ $\boldsymbol{x}$ および超パラメータ $\boldsymbol{\alpha}$ に対して独立なので

$$\begin{aligned} P(k|\boldsymbol{x};\boldsymbol{\alpha}) &= \int P(k|\boldsymbol{\theta}) P(\boldsymbol{\theta}|\boldsymbol{x};\boldsymbol{\alpha}) d\boldsymbol{\theta} \\ &= \int \theta_k P(\boldsymbol{\theta}|\boldsymbol{x};\boldsymbol{\alpha}) d\boldsymbol{\theta} \\ &= \mathbb{E}[\theta_k|\boldsymbol{x};\boldsymbol{\alpha}] \end{aligned} \tag{2.20}$$

$\boldsymbol{\theta}$ の分布は式 (2.17) のディリクレ分布で，その平均は式 (2.14) で与えられるので

$$P(k|\boldsymbol{x};\boldsymbol{\alpha}) = \frac{x_k + \alpha_k}{\sum_i (x_i + \alpha_i)} \tag{2.21}$$

となる．

2.1.2 グラフィカルモデル

確率的グラフィカルモデル (probabilistic graphical model) は確率分布の図式的な表現である．ここではグラフのリンクが特定の方向性を持ち矢印で書かれるベイジ

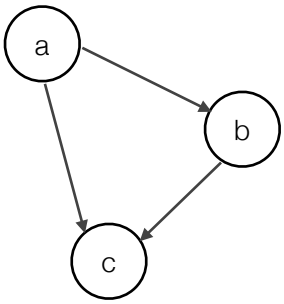

図 2.1: 3 変数 a, b, c の同時確率分布を表現する有向グラフィカルモデル

アンネットワーク (Bayesian network)（有向グラフィカルモデル (directed graphical model) とも呼ばれる）について述べる．

3 つの確率変数 a, b, c の上の同時分布 $P(a, b, c)$ を考える．確率の乗法定理を繰り返し適用することにより，同時分布は

$$\begin{aligned} P(a, b, c) &= P(c|a, b)P(a, b) \\ &= P(c|a, b)P(b|a)P(a) \end{aligned} \tag{2.22}$$

と書ける．式 (2.22) の右辺をグラフィカルモデルで表現する．まず確率変数 a, b, c に対応するノードを描き，各ノードとそのノード変数上の条件付き分布とを対応させる．そして，各条件付き分布に対応するノードに向かって，条件付けられた変数ノードからの有向リンクを付与する．例えば $P(c|a, b)$ に対応するノード c にはノード a, b からのリンクが付与される．一方，$P(a)$ に対応するノードに向かうリンクはない．その結果得られるグラフは図 2.1 のようになる．

他の例として，変数 $\boldsymbol{x}$ が式 (2.5) の多項分布に従い，そのパラメータ $\boldsymbol{\theta}$ が式 (2.11) のディリクレ分布に従うとき，同時分布は

$$\begin{aligned} P(\boldsymbol{x}, \boldsymbol{\theta}; \boldsymbol{\alpha}) &= P(\boldsymbol{x}|\boldsymbol{\theta}; \boldsymbol{\alpha})P(\boldsymbol{\theta}; \boldsymbol{\alpha}) \\ &\propto P(\boldsymbol{\theta}; \boldsymbol{\alpha}) \prod_{k=1}^{K} P(x_k|\boldsymbol{\theta}) \end{aligned} \tag{2.23}$$

$$P(x_k|\boldsymbol{\theta}) = \theta_k^{x_k} \tag{2.24}$$

となる．このグラフィカルモデルは図 2.2 のようになる．図 2.2 中の四角はプレート (plate) といい，x_k で代表される同質の一連の変数が K 個あることを表している．また $\boldsymbol{\alpha}$ を表すノードは小さな黒丸となっているが，これはパラメータを意味する．

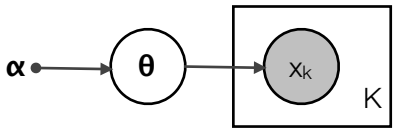

図 2.2: 変数 $\boldsymbol{x}, \boldsymbol{\theta}$ の同時確率分布を表現する有向グラフィカルモデル

最後にノード x_k がグレーとなっているのは，x_k だけが観測できる変数 (observed variable) であることを表現している．

2.1.3 サンプリング

サンプリングとはある確率分布 $P(\boldsymbol{z})$ に従い独立に抽出された，確率変数 $\boldsymbol{z}$ の値のサンプルの集合 $\boldsymbol{z}^{(l)}(l = 1, \ldots, L)$ を得ることである．サンプリングにより得たサンプルを使って，変数 $\boldsymbol{z}$ の予測分布を得たり，$\boldsymbol{z}$ の関数 $f(\boldsymbol{z})$ の期待値を予測したりすることが可能である．本節ではギブスサンプリング (Gibbs sampling) について述べる．

ギブスサンプリング [AA84][Bis06] はマルコフ連鎖モンテカルロ (Marcov chain Monte Carlo) によるサンプリング法の一種で，Metropolis-Hastings アルゴリズムの特別な場合とみなすことができる．

サンプリングしたい確率分布 $P(\boldsymbol{z}) = P(z_1, \ldots, z_M)$ を考え，マルコフ連鎖のある初期状態を選択したと仮定する．ギブスサンプリングの各ステップでは，１つの変数の値が更新される．その際，他の変数の値を固定した条件での，更新する変数の条件付き分布（完全条件付き分布，full conditional probability）に従って抽出した値で置き換える．すなわち，z_i を更新するとき，分布 $P(z_i|\boldsymbol{z}_{-i})$ から抽出した値で置き換える．ここで $\boldsymbol{z}_{-i}$ は $\boldsymbol{z}$ から z_i を除いたものを表す．この手続きは，各ステップで更新する変数を，ある決まった順序で循環するか，何らかの分布にしたがってランダムに選択して繰り返される．

例えば3つの確率変数 (z_1, z_2, z_3) を扱い，アルゴリズムのステップ τ で値 $z_1^{(\tau)}, z_2^{(\tau)}, z_3^{(\tau)}$ を得ているとする．まず，z_1 を条件付き分布

$$P(z_1|z_2^{(\tau)}, z_3^{(\tau)}) \tag{2.25}$$

からサンプリングして得た新しい値 $z_1^{(\tau+1)}$ で置き換える．次に，新しい z_1 の値を以

降のサンプリングのステップでそのまま用いて，z_2 を条件付き分布

$$P(z_2|z_1^{(\tau+1)}, z_3^{(\tau)}) \tag{2.26}$$

からサンプリングして得た値 $z_2^{(\tau+1)}$ で置き換える．そして z_3 を

$$P(z_3|z_1^{(\tau+1)}, z_2^{(\tau+1)}) \tag{2.27}$$

から抽出したサンプル $z_3^{(\tau+1)}$ で更新する．このステップを実行することによりサンプル $z_1^{(\tau+1)}, z_2^{(\tau+1)}, z_3^{(\tau+1)}$ を得る．

各条件付き分布からの抽出は，区間 $(0, 1)$ で一様分布する擬似乱数 y を発生させて行う．例えば，z_1 が連続な値をとる確率変数で，$z_1 \in (-\infty, \infty)$ ならば，累積分布関数 (cumulative distribution function)

$$\int_{-\infty}^{z_1} P(\hat{z}_1|z_2^{\tau}, z_3^{\tau}) d\hat{z}_1 \tag{2.28}$$

の値との比較を行う．すなわち

$$\int_{-\infty}^{z_1} P(\hat{z}_1|z_2^{\tau}, z_3^{\tau}) d\hat{z}_1 = y \tag{2.29}$$

を満たす z_1 をもって $z_1^{(\tau+1)}$ とする．

2.2 確率的言語モデルと文書検索

本節では確率的言語モデルと，それに基づく文書検索の技術について触れる．

2.2.1 言語モデルに基づく文書検索

言語を数学的に表現するとき，記号列の生起する確率（起こりやすさ）を考慮した確率的な言語を考えることができる．記号の有限集合 Σ に対し，Σ 上の確率的言語 (probabilistic language) を 2 項組 (L, P) により定義する．ここで L は Σ 上の言語であり，P は Σ^* から $[0, 1]$ への実数値関数（確率関数）である．ただし Σ^* は Σ のクリーネ閉包である．また，関数 P は次の条件を満たす．

- $x \in \Sigma^*$ に対し， $x \notin L \Rightarrow P(x) = 0$
- $x \in \Sigma^*$ に対し， $x \in L \Rightarrow 0 \leq P(x) \leq 1$
- $\displaystyle\sum_{x \in L} P(x) = 1$

確率的言語により，文あるいは単語列，文字列などに対して，それらが起こる確率を考えることができる．これらの確率を与えるモデルのことを確率的言語モデル (probabilistic language model) あるいは単に言語モデル (language model) と呼ぶ [北 99].

この言語モデルを文書検索に応用することができる [PC98]．文書検索に応用する際は，１つ１つの文書ごとにその文書を生成したモデル（文書モデル）を考える．このときある文書とクエリとの適合度は，その文書を生成した文書モデルがクエリを生成する尤度で与えられる．すなわち，クエリ $\boldsymbol{r}$ に対する文書 $\boldsymbol{d}$ の適合度は $P(\boldsymbol{r}|\boldsymbol{d})$ で与えられる．標準的な文書検索モデルでは，1-gram モデルを採用する．すなわち，文書を bag of words と考え，さらに語の生成は独立であると仮定するので，$P(\boldsymbol{r}|\boldsymbol{d})$ は

$$P(\boldsymbol{r}|\boldsymbol{d}) = \prod_{w \in \boldsymbol{r}} P(w|\boldsymbol{d}) \tag{2.30}$$

と表せる．$P(w|\boldsymbol{d})$ は $\boldsymbol{d}$ より推定する．文書 $\boldsymbol{d}$ 中に単語 w が $\#(w, \boldsymbol{d})$ 回出現したとき，$P(w|\boldsymbol{d})$ の最尤推定値は

$$P_{\mathrm{ml}}(w|\boldsymbol{d}) = \frac{\#(w, \boldsymbol{d})}{|\boldsymbol{d}|} \tag{2.31}$$

である．ここで $|\boldsymbol{d}|$ は $\boldsymbol{d}$ に含まれる語の総数（文書長）である．

推定のためのサンプル数 $|\boldsymbol{d}|$ は小さいので，$P(w|\boldsymbol{d})$ の分散が大きい．特に $\boldsymbol{d}$ 中に出現しなかった語は生成確率が 0 となってしまう．これに対処するため，推定値の補正を行うことをスムージング (smoothing) という．

検索対象の文書集合を C とする．よく用いられるスムージング法では，背景分布 (background distribution)

$$P_{\mathrm{ml}}(w|C) = \frac{\#(w, C)}{|C|} \qquad (2.32)$$
$$|C| = \sum_{\boldsymbol{d}\in C} |\boldsymbol{d}|$$

との凸結合 (convex combination) を行う．

$$P(w|\boldsymbol{d}) = \lambda P_{\mathrm{ml}}(w|\boldsymbol{d}) + (1-\lambda) P_{\mathrm{ml}}(w|C) \qquad (2.33)$$

ここで $\lambda \geq 0$ はパラメータである．これにより，少なくとも文書集合に出現した語に対しては確率 0 を付与することはなくなる．また,「文書 $\boldsymbol{d}$ が属する文書集合」という，$\boldsymbol{d}$ と無関係でない量により，推定値の分散を抑えることができる．ただしこの推定量は不偏推定量 (unbiased estimator) ではないことに注意する必要がある．パラメータ λ を大きくすると，分散は大きいが，データとして $\boldsymbol{d}$ を重視する．一方小さくすると，分散は小さくなるが，推定量へのバイアスが大きくなる．

特にサンプル数 $|\boldsymbol{d}|$ を考慮して凸結合を行うと

$$P(w|\boldsymbol{d}) = \frac{|\boldsymbol{d}|}{|\boldsymbol{d}| + \lambda} P_{\mathrm{ml}}(w|\boldsymbol{d}) + \frac{\lambda}{|\boldsymbol{d}| + \lambda} P_{\mathrm{ml}}(w|C) \qquad (2.34)$$

となる．これは文書長 $|\boldsymbol{d}|$ が大きいほど $P_{\mathrm{ml}}(w|\boldsymbol{d})$ の寄与が大きくなる．$|\boldsymbol{d}|$ が大きいほど $P_{\mathrm{ml}}(w|\boldsymbol{d})$ の分散は小さくなるので，合理的なスムージングと言える．式 (2.34) はディリクレスムージング (Dirichlet smoothing) と呼ばれる．

ディリクレスムージングについて，ベイズ主義の立場から補足する．
語の出現の事前分布をディリクレ分布 $\mathcal{DIR}(\boldsymbol{\alpha})$ とすると，予測分布 $P(w|\boldsymbol{d})$ は次のようになる．

$$P(w|\boldsymbol{d}) = \frac{\#(w, \boldsymbol{d}) + \alpha_w}{|\boldsymbol{d}| + \sum \alpha} \qquad (2.35)$$

α_w は語 w の有効観測数である．

ここで超パラメータ $\boldsymbol{\alpha}$ を文書集合 C における相対頻度 $P_{\mathrm{ml}}(w|C)$ の定数倍とする．すなわち

$$\alpha_w = \lambda P_{\mathrm{ml}}(w|C) \tag{2.36}$$

とすると，式 (2.35) は

$$\begin{aligned} P(w|\boldsymbol{d}) &= \frac{\#(w,\boldsymbol{d}) + \lambda P_{\mathrm{ml}}(w|C)}{|\boldsymbol{d}| + \sum \lambda P_{\mathrm{ml}}(w|C)} \\ &= \frac{\#(w,\boldsymbol{d}) + \lambda P_{\mathrm{ml}}(w|C)}{|\boldsymbol{d}| + \lambda} \\ &= \frac{\#(w,\boldsymbol{d})}{|\boldsymbol{d}| + \lambda} + \frac{\lambda}{|\boldsymbol{d}| + \lambda} P_{\mathrm{ml}}(w|C) \\ &= \frac{|\boldsymbol{d}|}{|\boldsymbol{d}| + \lambda} \frac{\#(w,\boldsymbol{d})}{|\boldsymbol{d}|} + \frac{\lambda}{|\boldsymbol{d}| + \lambda} P_{\mathrm{ml}}(w|C) \\ &= \frac{|\boldsymbol{d}|}{|\boldsymbol{d}| + \lambda} P_{\mathrm{ml}}(w|\boldsymbol{d}) + \frac{\lambda}{|\boldsymbol{d}| + \lambda} P_{\mathrm{ml}}(w|C) \end{aligned} \tag{2.37}$$

となり，式 (2.34) と等しくなる．

2.2.2 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA)[BNJ03] は文書コーパスのような離散データ（質的データ）の集合の確率的生成モデルである．LDA における文書モデルとは，まず語の生成確率分布を与えるモデルとして複数のトピックを考え，そして文書をトピックの混合モデル (mixture model) と考える，というものである．すなわち，まず，その文書で表現したい内容によって，どのトピックが好まれるかが決まり，次に，好みに従って選ばれたトピックを表現する単語がランダムに選ばれる．文書モデルはトピックの finite mixture で表され，その混合重みで特徴付けられる (finite mixture については式 (3.1) を参照されたい)．トピックにおける語の生成確率分布によって，語の意味的な関連性を表現することが出来る．また文書をトピックの finite mixture とすることで文書を簡潔に記述できる．この特徴から，LDA は文書分類や要約などに利用されている．

まず文書 $\boldsymbol{d}$ は複数の語 $\{w_1, \ldots, w_{|\boldsymbol{d}|}\}$ からなる bag of words とみなす．語の生成モデルとして複数のトピックを考え，語はいずれかのトピックから生成されるとする．すなわち，文書中の各単語に対してそれを生成したトピックがあると考え

られる．したがって，観測できる文書中の語の他に，隠れ (latent) 変数として１つ１つの語を生成したトピックがある．また，トピックからの語の生成確率分布はトピックごとに異なるとする．これにより，語の意味的な背景を表現できる．例えば,「音楽」や「絵画」という語の生成確率が高いトピックは「芸術」というトピックだと考えることが出来,「学校」や「教師」といった単語の生成確率が高いトピックは「教育」というトピックだと考えることが出来る．

各文書はトピックの finite mixture として表現される．従って，各文書は各トピックの重み＝選択される確率の分布で特徴付けられる．例えば，芸術について書かれた文書ならば「芸術」というトピックの選択される確率（重み）が高いと考えられる．

以上から，K 個のトピック $\{t_1, \ldots, t_K\}$，N 個の語彙 $\{v_1, \ldots, v_N\}$ のもとで，M 個の文書からなる文書集合 $\{\boldsymbol{d}_1, \ldots, \boldsymbol{d}_M\}$ の生成は次のようにモデル化される．

- for each $\boldsymbol{d}_i$
  1. ディリクレ分布 $\mathcal{DIR}(\boldsymbol{\alpha})$ に従ってトピックの分布のパラメータ $\boldsymbol{\theta}_i$ を生成する
  2. for each $w_{ij} \in \boldsymbol{d}_i$
     (a) 多項分布 $\mathcal{MULTI}(\boldsymbol{\theta}_i)$ に従ってトピックを選択する．選択されたトピックを t_k とする
     (b) $z_{ij} \leftarrow t_k$
     (c) 多項分布 $\mathcal{MULTI}(\boldsymbol{\phi}_k)$ によって単語を生成する．生成された単語を v_l とする
     (d) $w_{ij} \leftarrow v_l$

ここで w_{ij} は i 番目の文書の j 番目の語を表す確率変数であり，z_{ij} は w_{ij} を生成するトピックを表す確率変数である．また $\boldsymbol{\phi}_k$ はトピック t_k が語を生成する確率分布のパラメータであり，$\boldsymbol{\beta}$ をパラメータとするディリクレ分布 $\mathcal{DIR}(\boldsymbol{\beta})$ に従う．

この生成の過程をグラフィカルモデルで表すと図 2.3 のようになる．図からもわかるように，LDA は３段階の階層からなるモデルである．$\boldsymbol{\alpha}, \boldsymbol{\beta}$ はコーパスレベルのパラメータで，最初に決定されその後は不変である．$\boldsymbol{\theta}_i$ は文書レベルの変数で，文書ごとに１回生成される．z_{ij}, w_{ij} は単語レベルの変数で，各単語ごとに１回生成される．

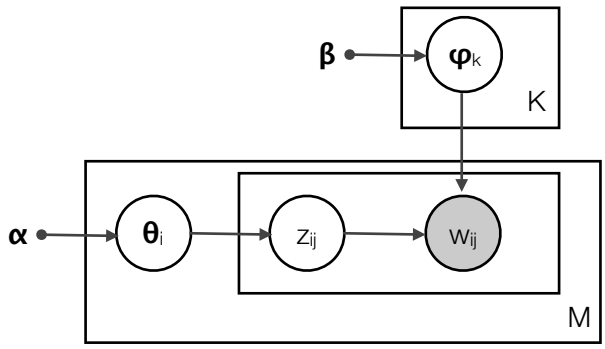

図 2.3: LDA のグラフィカルモデル

2.2.3 統計的機械翻訳と翻訳モデル

近年，機械翻訳の分野では，大規模な「対訳コーパス」から，自動的に翻訳機をつくる「統計的機械翻訳 (statistical machine translation, SMT) 」という技術が発展している．対訳コーパスとは，次のように，同じことを別の言語で書いた文のペアを大量に集めたものである．

- 英語：He drove the fire engine.
- 日本語：彼はその消防車を運転した.

このようなデータが大量にあれば，“fire engine” が「消防車」に翻訳される確率などが得られる．統計的機械翻訳の技術として，Brown らの IBM Model 1[BDPDPM93] を紹介する．Brown らは，翻訳元言語の文 $\boldsymbol{f}$ から翻訳先言語の文 $\boldsymbol{e}$ への翻訳に対して，次のようなモデルを想定している．

- 翻訳先言語の話者は文 $\boldsymbol{f}$ を発話あるいは記述するとき，頭の中で文 $\boldsymbol{e}$ を思い浮かべている．
- 翻訳とは出力された文 $\boldsymbol{f}$ から頭の中の文 $\boldsymbol{e}$ を復号する処理である．

このモデルのもとで，翻訳先の文 $\boldsymbol{e}$ の MAP 推定量は次のようになる．

$$\begin{aligned} \hat{\boldsymbol{e}} &= \operatorname*{argmax}_{\boldsymbol{e}} P(\boldsymbol{e}|\boldsymbol{f}) \\ &= \operatorname*{argmax}_{\boldsymbol{e}} P(\boldsymbol{f}|\boldsymbol{e})P(\boldsymbol{e}) \end{aligned} \tag{2.38}$$

したがって，統計的機械翻訳は

1. 言語モデル確率 $P(\boldsymbol{e})$ の推定
2. 翻訳モデル確率 $P(\boldsymbol{f}|\boldsymbol{e})$ の推定
3. 最尤推定単語列 $\hat{\boldsymbol{e}}$ の探索

という 3 つの部分問題に帰着される．Brown らは論文 [BDPDPM93] 中で部分問題 2 に対するモデルとして Model 1-5 を紹介しており，いずれも言語学的な知識を必要としない，EM アルゴリズムに基づくモデルである．中でも Model 1 はすべての単語対応付け (word alignment) を平等に扱う．Model 1 では翻訳先の語 e が翻訳元の語 f に翻訳される確率 $P(f|e)$ を必要とし，対訳コーパス $\{(\boldsymbol{f}^{(1)}|\boldsymbol{e}^{(1)}), \ldots, (\boldsymbol{f}^{(N)}|\boldsymbol{e}^{(N)})\}$ からは以下のように得られる．

$$P(f|e) = \frac{1}{\lambda_e} \sum_{i=1}^{N} c(f|e; \boldsymbol{f}^{(i)}, \boldsymbol{e}^{(i)}) \tag{2.39}$$

ここで λ_e は正規化のための要素である．また $c(f|e; \boldsymbol{f}^{(i)}, \boldsymbol{e}^{(i)})$ は

$$c(f|e; \boldsymbol{f}^{(i)}, \boldsymbol{e}^{(i)}) = \frac{P(f|e)}{P(f|e_1^{(i)}) + \cdots + P(f|e_n^{(i)})} \#(f, \boldsymbol{f}^{(i)}) \#(e, \boldsymbol{e}^{(i)}) \tag{2.40}$$

として得られる．ここで $\{e_1^{(i)}, \ldots, e_n^{(i)}\}$ は $\boldsymbol{e}^{(i)}$ に出現する単語の集合である．また $\#(f, \boldsymbol{f}^{(i)})$ は $\boldsymbol{f}^{(i)}$ 中に f が出現した頻度である．

式 (2.39)，式 (2.40) は再帰的に定義されている．従って変換確率に適当な初期値を割り当て，収束するまで処理を繰り返す．翻訳確率は，いかなる初期値でも必ず同じ値に収束することがわかっている．

Berger らはこのモデルを検索タスクに応用することを提案している [BL99]. Berger らはクエリ $\boldsymbol{r}$ と適合文書 $\boldsymbol{d}$ との関係を，文 $\boldsymbol{f}$ と $\boldsymbol{e}$ との関係に見立てている．すなわち

- ユーザはクエリ $\boldsymbol{r}$ を入力するとき，理想の適合文書 $\boldsymbol{d}$ を想定している．
- 検索とは理想の文書 $\boldsymbol{d}$ に近いものを探すタスクである．

というモデルを提案している．すると，クエリ $\boldsymbol{r}$ と文書 $\boldsymbol{d}$ との適合度は，$\boldsymbol{r}$ が $\boldsymbol{d}$ に翻訳される確率 $P(\boldsymbol{d}|\boldsymbol{r})$ となる．

$$P(\boldsymbol{d}|\boldsymbol{r}) \propto P(\boldsymbol{r}|\boldsymbol{d})P(\boldsymbol{d}) \tag{2.41}$$

であり，事前分布 $P(\boldsymbol{d})$ が一様分布とすると，適合度は $P(\boldsymbol{r}|\boldsymbol{d})$ となる．これは以下のように計算される．

$$P(\boldsymbol{r}|\boldsymbol{d}) = \prod_{w \in \boldsymbol{r}} P(w|\boldsymbol{d}) \tag{2.42}$$

$$P(w|\boldsymbol{d}) = \frac{|\boldsymbol{d}|}{|\boldsymbol{d}|+1} P_{\mathrm{tr}}(w|\boldsymbol{d}) + \frac{1}{|\boldsymbol{d}|+1} P(w|null) \tag{2.43}$$

$$P_{\mathrm{tr}}(w|\boldsymbol{d}) = \sum_{t \in \boldsymbol{d}} P(w|t) P_{\mathrm{ml}}(t|\boldsymbol{d}) \tag{2.44}$$

ここで $P(w|null)$ は語 w が文書 $\boldsymbol{d}$ 中の「偽の語」から翻訳される確率を表している．これは統計的機械翻訳において，翻訳先の文中のある語が翻訳元の文のどの語とも対応付けられない (alignment されない) ことを表現している．

ただし翻訳モデルには「自己翻訳」の問題がある．検索においては，クエリと検索対象文書は同じ言語であることが普通である．すると翻訳元の語と翻訳先の語が同じ語 w ということが起こる．このとき翻訳確率 $P(w|w)$ が低すぎると，マッチした語に対する重みが低くなり検索性能は低下する．逆に高すぎると翻訳モデルのメリットを活かせない．この問題に対処するためにいくつかのアプローチが提案されている．

2.3 Q&A アーカイブに関する研究

本章では Q&A アーカイブの情報源としての利用に関する研究を紹介する．

2.3.1 コミュニティベース Q&A の情報の質

Harper らはウェブ Q&A サイトの回答の質について次のように報告している [HRRK08].

- 無料の Q&A サイトより回答者に報酬のあるサイトの方が回答の質がよく，支払う報酬が高いほど，長くよい回答が得られる．
- 特定の個人が答えるより，多くの人間が回答できるシステムの方が，多様な回答を得られ，またすぐに回答がある．

また Harper らは Q&A サイトの質問を “informational” と “conversational” とに分類している [HMK09].「ミャンマーとビルマの違いは何ですか.」というような，情報を得る目的でなされたものを “informational” と定義し,「進化論を信じますか.」というような，議論を引き起こすのが目的の質問を “conversational” と定義している．“conversational” な質問においては，アーカイブし利用していく価値のあるものはほとんどないと報告している．また機械学習によりこれらを自動で分類する実験を行い，89.7%の精度で分類することができたと報告している．

Jeon らはクリック回数などの non-textual な特徴量を用いて回答の質を予測するフレームワークを提案している [JCLP06]．特徴量として質問者による評価，ページの印刷回数，クリック回数などを用い，カーネル密度推定を施した後，回答の質との相関を調べている．その結果，回答が ”best answer” に選ばれることの多い回答者による回答は質が良い，などの結果を得ている．さらにこれを類似質問検索に組み入れ，質問が類似しているだけでなく回答の質も高いものを検索することに成功している．

2.3.2 コミュニティベース Q&A を利用した研究

森らは Q&A サイトのデータを用いた質問応答の研究を行っている [MSI08]．この手法では，クエリ質問文と似た書き方の質問文をデータベースから抽出し，それに対する回答文から回答の手がかり表現を抽出する．そしてクエリ質問文のキー

ワードと抽出した手がかり表現を用いてウェブを通じて回答する．non-factoid 型の質問であっても分類を必要としないのが利点である．

2.3.3 類似質問検索

類似質問検索においては，クエリの語と検索対象のアーカイブ内の質問文の語とのミスマッチが問題となる．Jeon らはこの問題を解決するために翻訳モデルを用いている [JCL05]．類似した質問のペアの集合を対訳コーパスとし，翻訳確率を学習する．「2 つの Q&A ペアの回答文が類似していれば，質問文も類似している」という仮定のもと，類似質問ペアを収集する．学習した翻訳確率 $P(w|t)$ を用いて，スコアは次のようにして計算される．

$$P(\boldsymbol{r}|\boldsymbol{d}) = \prod_{w \in \boldsymbol{r}} P(w|\boldsymbol{d}) \tag{2.45}$$

$$P(w|\boldsymbol{d}) = (1-\lambda) \sum_{t \in \boldsymbol{q}} P(w|t) P_{\mathrm{ml}}(t|\boldsymbol{d}) + \lambda P_{\mathrm{ml}}(w|C) \tag{2.46}$$

ここで λ はパラメータである．background smoothing を行っている点は元の翻訳モデルとは異なる．また，自己翻訳問題に対処するため，自己翻訳確率 $P(w|w) = 1$ としている．

Xue らは Jeon らの手法を拡張し，翻訳モデルを言語モデルに統合した translation-based language model を提案している [XJC08]．translation-based language model は言語モデルと翻訳モデルの finite mixture を行うことで自己翻訳問題に対処している．さらに回答文の言語モデルを組み入れることで良い精度を得たとしている．質問文 $\boldsymbol{q}$, 回答文 $\boldsymbol{a}$ からなる Q&A ペア $(\boldsymbol{q}, \boldsymbol{a})$ における Xue らの検索モデルは以下のように与えられる．

$$P(w|(\boldsymbol{q}, \boldsymbol{a})) = \frac{|(\boldsymbol{q}, \boldsymbol{a})|}{|(\boldsymbol{q}, \boldsymbol{a})| + \lambda} P_{\mathrm{mx}}(w|(\boldsymbol{q}, \boldsymbol{a})) + \frac{\lambda}{|(\boldsymbol{q}, \boldsymbol{a})| + \lambda} P_{\mathrm{ml}}(w|C) \tag{2.47}$$

$$P_{\mathrm{mx}}(w|(\boldsymbol{q}, \boldsymbol{a})) = \alpha P_{\mathrm{ml}}(w|\boldsymbol{q}) + \beta \sum_{t \in \boldsymbol{q}} P(w|t) P_{\mathrm{ml}}(t|\boldsymbol{q}) + \gamma P_{\mathrm{ml}}(w|\boldsymbol{a}) \tag{2.48}$$

ここで $|(\boldsymbol{q}, \boldsymbol{a})| = |\boldsymbol{q}| + |\boldsymbol{a}|$ である．また，$\alpha, \beta, \gamma, \lambda$ はパラメータで，$\alpha + \beta + \gamma = 1$ である．翻訳確率 $P(w|t)$ は Q&A アーカイブを対訳コーパスとして扱うことで得ている．この手法は様々な確率モデルを組み合わせることにより性能の向上に成功している．しかしパラメータ $\alpha, \beta, \gamma, \lambda$ は人手で実験的に決定しなければならない

ため，労力を要する．また，実験的に決定するには，正解セットが既に与えられている必要がある．その他，新たなモデルを追加したり，既存のモデルを変更・削除したりするたびにパラメータ調整が必要である点も考慮するべきである．

Wang らは質問文の構文的特徴を扱うため，構文木 (syntactic tree) の構造に基づいた検索フレームワークを提案している [WMC09]．構文構造を表現する方法としてツリーカーネル関数 (tree kernel function) があるが，それでは厳格すぎるとし，論文中で構文木のマッチング法を新たに導入している．これは tree fragment の重みを，ノードの品詞やサイズ（含むノードの数），深さにより定義する．また部分的なマッチングを許している．これにより文法誤りに対してロバストな性能を発揮している．加えて， WordNet[FEL98] を利用して名詞及び動詞の意味的な類似度を計算し，ノードのマッチングスコアに反映させている．この手法はトレーニングの必要がなく，文法誤りに対してロバストであるという長所がある．しかし，文法誤りを含む場合は bag of words の検索モデルのほうが性能が良いとしている．

第 3 章
提案手法

3.1 概要

Q&A アーカイブとは，質問と回答のペアが大量に蓄積されたものである．アーカイブ全体を C とする．$C = \{(\boldsymbol{q}, \boldsymbol{a})_1, (\boldsymbol{q}, \boldsymbol{a})_2, \ldots, (\boldsymbol{q}, \boldsymbol{a})_L\}$ と表せる．C に含まれる質問の集合を Q とする．$Q = \{\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_M\}$ と表せる．C に含まれる回答の集合を A とする．$A = \{\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_N\}$ と表せる．すべての $(\boldsymbol{q}, \boldsymbol{a}) \in C$ に対して，$\boldsymbol{q} \in Q$, $\boldsymbol{a} \in A$ である．また，1つの質問に複数の回答がなされることや，1つの回答が複数の質問に対応する可能性を考慮すると，$M \leq L, N \leq L$ である．

類似質問検索とは，$(\boldsymbol{q}, \boldsymbol{a}) \in C$ を，クエリ質問 $\boldsymbol{r}$ に対する適合度により順位付けるタスクである．本研究では，検索のモデルとして，言語モデルによる検索を拡張した手法を提案する．すなわち，アーカイブ内の Q&A ペア $(\boldsymbol{q}, \boldsymbol{a})$ と $\boldsymbol{r}$ との適合度を，Q&A ペア $(\boldsymbol{q}, \boldsymbol{a})$ に対するモデルがクエリ $\boldsymbol{r}$ を生成する尤度 $P(\boldsymbol{r}|(\boldsymbol{q}, \boldsymbol{a}))$ で表現する．以降，文書を bag of words として扱い，文書中の各語の生成は独立であると仮定する．$P(\boldsymbol{r}|(\boldsymbol{q}, \boldsymbol{a}))$ は

$$P(\boldsymbol{r}|(\boldsymbol{q}, \boldsymbol{a})) = \prod_{w \in \boldsymbol{r}} P(w|(\boldsymbol{q}, \boldsymbol{a})) \tag{3.1}$$

として求められる．

すると問題は，Q&A ペアの生成をどのようにモデル化するか，ということになる．既存研究においても示されているように，クエリ質問と Q&A ペアとのマッチングを様々な側面から考えることで，性能が向上する．そこで本研究では，様々な確率的モデルを統合的に扱うフレームワークを提案する．すなわち，Q&A ペアを様々な情報に基づくモデルの混合 (mixture model) [MP00] として表現する．

K 個の基本モデル (mixture components) $M_1, M_2, \ldots, M_K$ による語の生成確率分布

$$P(w|(\boldsymbol{q}, \boldsymbol{a}), M_1), P(w|(\boldsymbol{q}, \boldsymbol{a}), M_2), \ldots, P(w|(\boldsymbol{q}, \boldsymbol{a}), M_K)$$

を考える．混合モデルは，係数 $\{c_1, c_2, \ldots, c_K\}$ を用いた finite mixture

$$P(w|(\boldsymbol{q}, \boldsymbol{a})) = \sum_{k=1}^{K} c_k P(w|(\boldsymbol{q}, \boldsymbol{a}), M_k) \tag{3.2}$$

$$\text{where} \quad c_k \geq 0, \ \sum_{k=1}^{K} c_k = 1$$

として与えることができる．すると，混合モデルの計算のためには次の 2 点を明

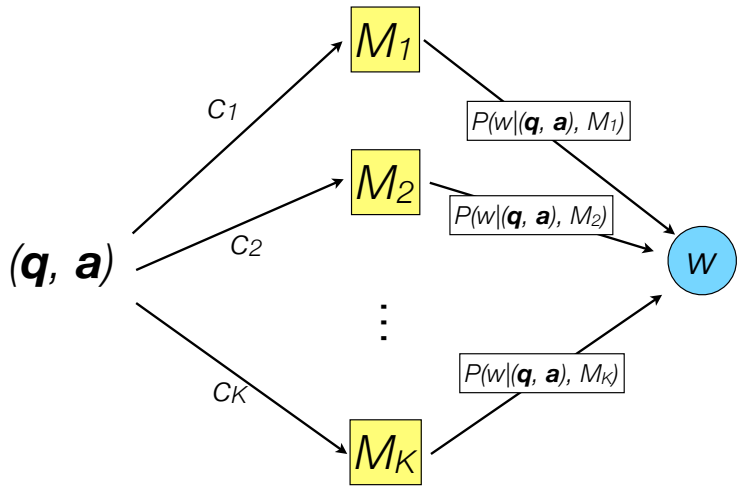

図 3.1: 混合モデルの概念図

らかにしなければならない．

- どのような基本モデルを採用するか
- 混合比率 c_k をどのように決定するか

基本モデルについては 3.4 節で述べる．Xue らの手法 [XJC08] において用いられているモデルをベースに，類似質問検索に適したものを検討する．

混合比率に関しては，まずその解釈・扱いについてここで検討する．

3.1.1 混合比率

混合比率 c_k は，語 w の生成にモデル M_k が使われる確率とみなすことができる．すなわち語の生成は，

1. 多項分布に従い，基本モデル $M_1, M_2, \ldots, M_K$ からモデルが 1 つ選ばれる
2. 選ばれた基本モデルから語が生成される

という過程と解釈できる．この概念図を図 3.1 に示す．

ここで，モデル選択の多項分布及び混合比率について，2 つの立場が考えられる．まずアーカイブ全体で一定，すなわち Q&A ペアによらないという立場がある．2 つ目は LDA のように，Q&A ペアによって異なるとする立場である．前者

の立場では，混合モデルは

$$P(w|(\boldsymbol{q},\boldsymbol{a})_i) = \sum_{k=1}^{K} c_k P(w|(\boldsymbol{q},\boldsymbol{a})_i, M_k) \tag{3.3}$$

となり，混合比率 c_k は Q&A ペアの番号 i によらない．一方後者の立場では，混合モデルは

$$P(w|(\boldsymbol{q},\boldsymbol{a})_i) = \sum_{k=1}^{K} c_{ik} P(w|(\boldsymbol{q},\boldsymbol{a})_i, M_k) \tag{3.4}$$

となり，混合比率 c_{ik} は Q&A ペアの番号 i による．

いずれの場合でも，本手法では，混合比率はモデルの予測分布として推定する．混合モデル $P(w|(\boldsymbol{q},\boldsymbol{a}))$ により文書を生成するためには，まず多項分布に従って基本モデルを選択するが，このモデル選択の多項分布からサンプリングを行い，得られたサンプルに基づき予測分布を計算し，その値をもって混合比率とするということである．サンプリングには 2.1.3 節で紹介したギブスサンプリングを用いる．

混合比率の推定は 2 節に分けて説明する．3.2 節で，モデル分布が Q&A によらない場合の混合比率推定を説明する．3.3 節で，モデル分布が Q&A ごとに異なる場合の混合比率推定を説明する．

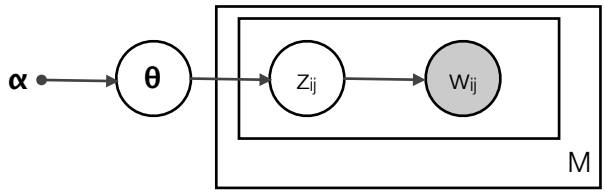

図 3.2: モデルの分布が等しい場合のグラフィカルモデル．そのパラメータ $\boldsymbol{\theta}$ は最初にサンプリングされ，以降は一定である

3.2 Q&A ペアに依存しない混合比率の推定

まず，文書集合の生成をモデル化する．

混合モデル $P(w|(\boldsymbol{q},\boldsymbol{a})_i)$ によって文書 $\boldsymbol{d}_i$ が生成されるとする．文書 $\boldsymbol{d}_i$ の語 w_{ij} を生成した基本モデルを変数 z_{ij} で表す．基本モデル z_{ij} は，$\{M_1, M_2, \ldots, M_K\}$ から多項分布 $\mathcal{MULTI}(\boldsymbol{\theta})$ に従ってランダムに選択される．この多項分布のパラメータ $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$ は Q&A ペアによらず一定である．パラメータ $\boldsymbol{\theta}$ はプロセスの最初に，ディリクレ事前分布 $\mathcal{DIR}(\boldsymbol{\alpha})$ に従ってランダムに選択されるものとする．ここで $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_K)$ は事前分布のパラメータである．

文書集合 $D = \{\boldsymbol{d}_i\}_{i=1}^{M}$ とする．文書集合 D の生成プロセスは以下のようになる．

1. ディリクレ分布 $\mathcal{DIR}(\boldsymbol{\alpha})$ に従って多項分布パラメータ $\boldsymbol{\theta}$ を生成する
2. for each $\boldsymbol{d}_i \in D$
   - for each $w_{ij} \in \boldsymbol{d}_i$
     (a) 多項分布 $\mathcal{MULTI}(\boldsymbol{\theta})$ に従ってモデルを選択し，z_{ij} に代入する
     (b) モデル z_{ij} に従って語 w_{ij} を生成する

このプロセスのグラフィカルモデルは図 3.2 のようになる．

3.2.1 予測分布

この生成プロセスにおいて，混合比率 c_k は，モデルの予測分布として得られる．文書集合 D が得られたときのモデルの予測分布とは，新たな語を生成するための

モデル z_0 の予測分布であり，$z_0 = M_k$ となる確率は

$$\begin{aligned} P(z_0 = M_k|Z;\boldsymbol{\alpha}) &= \int P(z_0 = M_k, \boldsymbol{\theta}|Z;\boldsymbol{\alpha})d\boldsymbol{\theta} \\ &= \int P(z_0 = M_k|\boldsymbol{\theta})P(\boldsymbol{\theta}|Z;\boldsymbol{\alpha})d\boldsymbol{\theta} \\ &= \int \theta_k P(\boldsymbol{\theta}|Z;\boldsymbol{\alpha})d\boldsymbol{\theta} \\ &= \mathbb{E}[\theta_k|Z;\boldsymbol{\alpha}] \end{aligned} \tag{3.5}$$

である．ここで

$$Z = \{z_{ij}\}_{ij} \tag{3.6}$$

である．
式 (2.21) と同様に，予測分布は

$$P(z_0 = M_k|Z;\boldsymbol{\alpha}) = \frac{\#M_k + \alpha_k}{\sum_k(\#M_k + \alpha_k)} \tag{3.7}$$

となる．ここで $\#M_k$ は文書集合 D 内でのモデル M_k の出現頻度である．
以上から，混合比率 c_k は

$$c_k = \frac{\#M_k + \alpha_k}{\sum_k(\#M_k + \alpha_k)} \tag{3.8}$$

として推定できる．

式 (3.8) の計算のためには，Z のサンプリングが必要である．サンプリングについて，以下の 2 点を説明する．

- 完全条件付き分布の導出
- ギブスサンプリングのアルゴリズム

3.2.2 完全条件付き分布

ギブスサンプリングのためには，完全条件付き分布が必要である．モデル z_{ij} の条件付き分布は,「文書集合 D の全語 W と，z_{ij} 以外のモデル Z_{-ij} がわかっているときの z_{ij} の分布」であり，例えば $z_{ij} = M_k$ となる確率は

$$P(z_{ij} = M_k|W, Z_{-ij};\boldsymbol{\alpha}) \tag{3.9}$$

と表される．以下に，完全条件付き確率の導出を行う．

まず，文書集合 D が生成される完全データの尤度は次のようになる．

$$P(W, Z, \boldsymbol{\theta}; \boldsymbol{\alpha}) = P(\boldsymbol{\theta}; \boldsymbol{\alpha}) \prod_{i=1}^{M} \prod_{j} P(z_{ij}|\boldsymbol{\theta}) P(w_{ij}|z_{ij}) \tag{3.10}$$

$z_{ij} = M_k$ のとき，$P(z_{ij}|\boldsymbol{\theta}) = \theta_k$ であるから

$$\begin{aligned} P(W, Z, \boldsymbol{\theta}; \boldsymbol{\alpha}) &= \left[\frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \right] \prod_{k=1}^{K} \theta_k^{\#M_k} \prod_{i,j} P(w_{ij}|z_{ij}) \\ &= \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \left[\prod_{k=1}^{K} \theta_k^{\#M_k + \alpha_k - 1} \right] \prod_{i,j} P(w_{ij}|z_{ij}) \end{aligned} \tag{3.11}$$

次に，多項分布に関して周辺化する．

$$\begin{aligned} P(W, Z; \boldsymbol{\alpha}) &= \int P(W, Z, \boldsymbol{\theta}; \boldsymbol{\alpha}) d\boldsymbol{\theta} \\ &= \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \int \prod_{k=1}^{K} \theta_k^{\#M_k + \alpha_k - 1} d\boldsymbol{\theta} \prod_{i,j} P(w_{ij}|z_{ij}) \\ &= \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \frac{\prod_k \Gamma(\#M_k + \alpha_k)}{\Gamma\left(\sum_k (\#M_k + \alpha_k)\right)} \prod_{i,j} P(w_{ij}|z_{ij}) \end{aligned} \tag{3.12}$$

ここで積分の計算には式 (2.13) を用いている．

さて，完全条件付き確率は次のようになる．

$$P(z_{ij} = M_k | W, Z_{-ij}; \boldsymbol{\alpha}) = \frac{P(z_{ij} = M_k, w_{ij} | W_{-ij}, Z_{-ij}; \boldsymbol{\alpha})}{P(w_{ij} | W_{-ij}, Z_{-ij}; \boldsymbol{\alpha})} \tag{3.13}$$

分母は k に依存しないので

$$\begin{aligned} P(z_{ij} = M_k | W, Z_{-ij}; \boldsymbol{\alpha}) &\propto P(z_{ij} = M_k, w_{ij} | W_{-ij}, Z_{-ij}; \boldsymbol{\alpha}) \\ &= \frac{P(W, Z; \boldsymbol{\alpha})}{P(W_{-ij}, Z_{-ij}; \boldsymbol{\alpha})} \end{aligned} \tag{3.14}$$

$P(W, Z; \boldsymbol{\alpha})$ および $P(W_{-ij}, Z_{-ij}; \boldsymbol{\alpha})$ は式 (3.12) で与えられるが，w_{ij}, z_{ij} を含めるか含めないかが異なる．さらにそのことにより $\#M_k$ の値が 1 変化することに注意する．すなわち，

$$P(W_{-ij}, Z_{-ij}; \boldsymbol{\alpha}) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \frac{\prod_{k'} \Gamma(\#M_{k'} + \alpha_{k'})}{\Gamma\left(\sum_k (\#M_k + \alpha_k)\right)} \prod_{(i',j') \neq (i,j)} P(w_{i'j'}|z_{i'j'}) \tag{3.15}$$

とおくと，

$$P(W,Z;\boldsymbol{\alpha}) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \frac{\Gamma(\#M_k + \alpha_k + 1) \prod_{k' \neq k} \Gamma(\#M_{k'} + \alpha_{k'})}{\Gamma(\sum_k (\#M_k + \alpha_k) + 1)} \prod_{i',j'} P(w_{i'j'}|z_{i'j'}) \quad (3.16)$$

となる．ここで和や積の変数を適当に書き換えている．
これにより

$$\frac{P(W,Z;\boldsymbol{\alpha})}{P(W_{-ij},Z_{-ij};\boldsymbol{\alpha})} = \frac{\Gamma(\#M_k + \alpha_k + 1)}{\Gamma(\#M_k + \alpha_k)} \frac{\Gamma(\sum_k (\#M_k + \alpha_k))}{\Gamma(\sum_k (\#M_k + \alpha_k) + 1)} P(w_{ij}|z_{ij}) \quad (3.17)$$

と計算される．
$z_{ij} = M_k$ であるから $P(w_{ij}|z_{ij}) = P(w_{ij}|(\boldsymbol{q},\boldsymbol{a})_i, M_k)$ となる．さらにガンマ関数の性質

$$\Gamma(x+1) = x\Gamma(x) \quad (3.18)$$

を用いると，完全条件付き確率は

$$P(z_{ij} = M_k|W,Z_{-ij};\boldsymbol{\alpha}) \propto \frac{\#M_k + \alpha_k}{\sum_k (\#M_k + \alpha_k)} P(w_{ij}|(\boldsymbol{q},\boldsymbol{a})_i, M_k) \quad (3.19)$$

となる．

得られた結果は 2 つの因子の積として書かれている．前の因子は式 (3.7) の右辺と同じである，しかし，式 (3.19) の $\#M_k$ は Z_{-ij} における計数であることに注意する必要がある．すなわち前の因子は，Z_{-ij} のもとでの予測分布

$$P(z_{ij} = M_k|Z_{-ij};\boldsymbol{\alpha}) \quad (3.20)$$

である．一方後ろの因子は，$z_{ij} = M_k$ をうけて，M_k が w_{ij} を生成する尤度である．すなわち選ばれるモデルは，Z_{-ij} の情報から選ばれる確率が高いと予測され，かつ w_{ij} を生成することが尤もらしいモデルである．

3.2.3 アルゴリズム

ギブスサンプリングでは，式 (3.19) を用いて確率変数 z_{ij} を更新する．ギブスサンプリングの $\tau+1$ 回目のサイクルのアルゴリズムはアルゴリズム 1 のようになる．ここで $z_{ij}^{(\tau)}$ は変数 z_{ij} の τ 回目のサイクルにおける値である．

ギブスサンプリングにおいて 1 つの変数が更新される際，変動があるのは，式

Algorithm 1 Gibbs sampling 1

for all $\boldsymbol{d}_i \in D$ **do**
  for all $w_{ij} \in \boldsymbol{d}_i$ **do**
    $\#z_{ij}^{(\tau)} \leftarrow \#z_{ij}^{(\tau)} - 1$
    choose $z_{ij}^{(\tau+1)} \sim P(z_{ij}|W, Z_{-ij}; \boldsymbol{\alpha})$
    $\#z_{ij}^{(\tau+1)} \leftarrow \#z_{ij}^{(\tau+1)} + 1$
  end for
end for

(3.19) の $\#M_k$ である．再び，この値は Z_{-ij} における計数であることに注意する．ある変数 z_{ij} が更新されるステップは，

1. まず，z_{ij} の値を見ないこととする（既知のデータから除外し，未知のものとする）．
   これに伴い，$\#z_{ij}^{(\tau)}$ の値を 1 減らす．

2. 式 (3.19) により，確率分布 $P(z_{ij}|W, Z_{-ij}; \boldsymbol{\alpha})$ を計算する．

3. $P(z_{ij}|W, Z_{-ij}; \boldsymbol{\alpha})$ に基づいてモデルを選択し $z_{ij}^{(\tau+1)}$ とする．
   これに伴い，$\#z_{ij}^{(\tau+1)}$ の値を 1 増やす．

となる．

このステップをすべての $z_{ij} \in Z$ に対して行うことを繰り返す．十分繰り返すことで，Z は初期状態から，尤もらしいものへ収束していく．初期状態はどうであっても収束には関係がないので，各モデルをランダムに割り当てることとする．

3.2.4 まとめ

Q&A ペアに依存しない混合比率の推定について，簡潔にまとめると次のようになる．

1. 文書集合 D を用意する
   D は混合比率推定の訓練データと言える．どのようなものが望ましいかは 4.2 節にて検証する．

2. 3.2.3 節のアルゴリズムにて十分なサンプリングを行う
   Z の初期状態は一様分布でよい．超パラメータ $\boldsymbol{\alpha}$ の調整については 4.3.2 節にて議論する．収束にかかるサイクル数は 4.2 節にて例を示している．

3. 式 (3.8) により混合比率を計算する

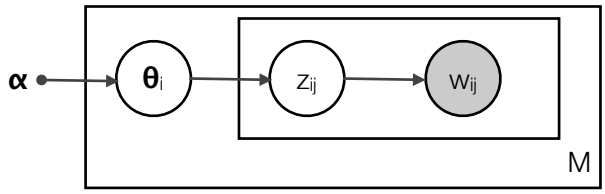

図 3.3: 文書ごとにモデルの分布が異なる場合のグラフィカルモデル．そのパラメータ $\boldsymbol{\theta}_i$ は文書ごとにサンプリングされる

3.3 Q&A ペアごとに異なる混合比率の推定

まず，文書集合の生成をモデル化する．notation は 3.2 節と同様である．

LDA と同様に，モデル選択の多項分布は，Q&A ペアごとにディリクレ分布 $\mathcal{DIR}(\boldsymbol{\alpha})$ に従ってランダムに選択されるものとする．文書集合 D を生成するために用いるモデル分布パラメータの集合を $\Theta = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_M\}$ と表す．ここで $\boldsymbol{\theta}_i = (\theta_{i1}, \theta_{i2}, \ldots, \theta_{iK})$ は文書 $\boldsymbol{d}_i$ を生成するために用いられるパラメータである．

文書集合 D の生成プロセスは以下のようになる．

- for each $\boldsymbol{d}_i \in D$
    1. ディリクレ分布 $\mathcal{DIR}(\boldsymbol{\alpha})$ に従って多項分布パラメータ $\boldsymbol{\theta}_i$ を生成する
    2. for each $w_{ij} \in \boldsymbol{d}_i$
        (a) 多項分布 $\mathcal{MULTI}(\boldsymbol{\theta}_i)$ に従ってモデルを選択し，z_{ij} に代入する
        (b) モデル z_{ij} によって w_{ij} を生成する

このプロセスのグラフィカルモデルは図 3.3 のようになる．

3.3.1 予測分布

混合比率 c_{ik} は，モデルの予測分布として推定できる．文書集合 D が得られたときのモデルの予測分布は，「Z がわかっているときに，文書 $\boldsymbol{d}_i$ に新たな語を付け加

えるためのモデル z_{i0} の分布」であり，たとえば $z_{i0} = M_k$ となる確率は

$$\begin{aligned} P(z_{i0} = M_k|Z;\boldsymbol{\alpha}) &= \int P(z_{i0} = M_k, \boldsymbol{\theta}_i|Z;\boldsymbol{\alpha})d\boldsymbol{\theta}_i \\ &= \int P(z_{i0} = M_k|\boldsymbol{\theta}_i)P(\boldsymbol{\theta}_i|Z;\boldsymbol{\alpha})d\boldsymbol{\theta}_i \\ &= \int \theta_{ik}P(\boldsymbol{\theta}_i|Z;\boldsymbol{\alpha})d\boldsymbol{\theta}_i \\ &= \mathbb{E}[\theta_{ik}|Z;\boldsymbol{\alpha}] \end{aligned} \tag{3.21}$$

となる．
式 (2.21) と同様に，予測分布は

$$P(z_{i0} = M_k|Z;\boldsymbol{\alpha}) = \frac{\#(M_k, \boldsymbol{d}_i) + \alpha_k}{\sum_{k=1}^{K}(\#(M_k, \boldsymbol{d}_i) + \alpha_k)} \tag{3.22}$$

となる．ここで $\#(M_k, \boldsymbol{d}_i)$ は文書 $\boldsymbol{d}_i$ におけるモデル M_k の出現頻度である．
以上から，混合比率 c_{ik} は

$$c_{ik} = \frac{\#(M_k, \boldsymbol{d}_i) + \alpha_k}{\sum_{k=1}^{K}(\#(M_k, \boldsymbol{d}_i) + \alpha_k)} \tag{3.23}$$

として推定できる．

式 (3.23) の計算のためには，Z のサンプリングが必要である．サンプリングについて，以下の 2 点を説明する．

- 完全条件付き分布の導出
- ギブスサンプリングのアルゴリズム

3.3.2 完全条件付き分布

式 (3.9) の完全条件付き確率の導出を行う．

まず，このプロセスで文書集合 D が生成される完全データの尤度 $P(W, Z, \Theta;\boldsymbol{\alpha})$ は次のようになる．

$$P(W, Z, \Theta;\boldsymbol{\alpha}) = \prod_{i=1}^{M} P(\boldsymbol{\theta}_i;\boldsymbol{\alpha}) \prod_{j} P(w_{ij}|z_{ij})P(z_{ij}|\boldsymbol{\theta}_i) \tag{3.24}$$

$z_{ij} = M_k$ ならば $P(z_{ij}|\boldsymbol{\theta}_i) = \theta_{ik}$ であるから

$$\begin{aligned} P(W, Z, \Theta; \boldsymbol{\alpha}) &= \prod_{i=1}^{M} \left[\frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_{ik}^{\alpha_k - 1} \right] \prod_{k=1}^{K} \theta_{ik}^{\#(M_k, \boldsymbol{d}_i)} \prod_j P(w_{ij}|z_{ij}) \\ &= \left[\frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \right]^M \prod_{i=1}^{M} \prod_{k=1}^{K} \theta_{ik}^{\#(M_k, \boldsymbol{d}_i) + \alpha_k - 1} \prod_j P(w_{ij}|z_{ij}) \end{aligned} \tag{3.25}$$

次に，モデル分布に関して周辺化する．

$$\begin{aligned} P(W, Z; \boldsymbol{\alpha}) &= \int P(W, Z, \Theta; \boldsymbol{\alpha}) d\Theta \\ &= \left[\frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \right]^M \prod_{i=1}^{M} \int \prod_{k=1}^{K} \theta_{ik}^{\#(M_k, \boldsymbol{d}_i) + \alpha_k - 1} d\boldsymbol{\theta}_i \prod_j P(w_{ij}|z_{ij}) \\ &= \left[\frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \right]^M \prod_{i=1}^{M} \frac{\prod_k \Gamma(\#(M_k, \boldsymbol{d}_i) + \alpha_k)}{\Gamma(\sum_k (\#(M_k, \boldsymbol{d}_i) + \alpha_k))} \prod_j P(w_{ij}|z_{ij}) \end{aligned} \tag{3.26}$$

完全条件付き確率は式 (3.14) で計算できるが，ここでも w_{ij}, z_{ij} の扱い及び $\#(M_k, \boldsymbol{d}_i)$ の値に変化が現れる．すなわち

$$\begin{aligned} P(W_{-ij}, Z_{-ij}; \boldsymbol{\alpha}) \propto & \frac{\prod_{k'} \Gamma(\#(M_{k'}, \boldsymbol{d}_i) + \alpha_{k'})}{\Gamma(\sum_k (\#(M_k, \boldsymbol{d}_i) + \alpha_k))} \prod_{j' \neq j} P(w_{ij'}|z_{ij'}) \\ & \times \prod_{i'=1, i' \neq i}^{M} \frac{\prod_k \Gamma(\#(M_k, \boldsymbol{d}_{i'}) + \alpha_k)}{\Gamma(\sum_k (\#(M_k, \boldsymbol{d}_{i'}) + \alpha_k))} \prod_j P(w_{i'j}|z_{i'j}) \end{aligned} \tag{3.27}$$

と表すと，

$$\begin{aligned} P(W, Z; \boldsymbol{\alpha}) \propto & \frac{\Gamma(\#(M_k, \boldsymbol{d}_i) + \alpha_k + 1) \prod_{k' \neq k} \Gamma(\#(M_{k'}, \boldsymbol{d}_i) + \alpha_{k'})}{\Gamma(\sum_k (\#(M_k, \boldsymbol{d}_i) + \alpha_k) + 1)} \prod_{j'} P(w_{ij'}|z_{ij'}) \\ & \times \prod_{i'=1, i' \neq i}^{M} \frac{\prod_k \Gamma(\#(M_k, \boldsymbol{d}_{i'}) + \alpha_k)}{\Gamma(\sum_k (\#(M_k, \boldsymbol{d}_{i'}) + \alpha_k))} \prod_j P(w_{i'j}|z_{i'j}) \end{aligned} \tag{3.28}$$

となる．ただし i, j に依存しない項は省略している．

Algorithm 2 Gibbs sampling 2

for all $\boldsymbol{d}_i \in D$ **do**
  for all $w_{ij} \in \boldsymbol{d}_i$ **do**
    $\#(z_{ij}^{(\tau)}, \boldsymbol{d}_i) \leftarrow \#(z_{ij}^{(\tau)}, \boldsymbol{d}_i) - 1$
    choose $z_{ij}^{(\tau+1)} \sim P(z_{ij}|W, Z_{-ij}; \boldsymbol{\alpha})$
    $\#(z_{ij}^{(\tau+1)}, \boldsymbol{d}_i) \leftarrow \#(z_{ij}^{(\tau+1)}, \boldsymbol{d}_i) + 1$
  end for
end for

以上から

$$
\begin{aligned}
P(z_{ij} = M_k|W, T_{-ij}; \boldsymbol{\alpha}) &\propto \frac{P(W, Z; \boldsymbol{\alpha})}{P(W_{-ij}, Z_{-ij}; \boldsymbol{\alpha})} \\
&= \frac{\Gamma(\#(M_k, \boldsymbol{d}_i) + \alpha_k + 1)}{\Gamma(\#(M_k, \boldsymbol{d}_i) + \alpha_k)} \frac{\Gamma(\sum_k(\#(M_k, \boldsymbol{d}_{i'}) + \alpha_k))}{\Gamma(\sum_k(\#(M_k, \boldsymbol{d}_i) + \alpha_k) + 1)} P(w_{ij}|z_{ij}) \\
&= \frac{\#(M_k, \boldsymbol{d}_i) + \alpha_k}{\sum_k(\#(M_k, \boldsymbol{d}_i) + \alpha_k)} P(w_{ij}|(\boldsymbol{q}, \boldsymbol{a})_i, M_k)
\end{aligned}
\tag{3.29}
$$

となる．式 (3.19) と異なるのは右辺の第 1 因子であり，式 (3.22) の右辺と等しい．すなわちこれもまた予測分布 $P(z_{i0} = M_k|Z_{-ij}; \boldsymbol{\alpha})$ である．選ばれるモデルは，Z_{-ij} の情報から選ばれる確率が高いと予測され，かつ w_{ij} を生成することが尤もらしいモデルである．

3.3.3 アルゴリズム

ギブスサンプリングでは，式 (3.29) を用いて確率変数 z_{ij} を更新する．ギブスサンプリングの $\tau+1$ 回目のサイクルのアルゴリズムはアルゴリズム 2 のようになる．

ギブスサンプリングにおいて 1 つの変数が更新される際，変動があるのは，式 (3.29) の $\#(M_k, \boldsymbol{d}_i)$ である．再び，この値は Z_{-ij} における計数であることに注意する．ある変数 z_{ij} が更新されるステップは，

1. まず，z_{ij} の値を見ないこととする（既知のデータから除外し，未知のものとする）．
   これに伴い，$\#(z_{ij}^{(\tau)}, \boldsymbol{d}_i)$ の値を 1 減らす．
2. 式 (3.29) により，確率分布 $P(z_{ij}|W, Z_{-ij}; \boldsymbol{\alpha})$ を計算する．

3. $P(z_{ij}|W, Z_{-ij}; \boldsymbol{\alpha})$ に基づいてモデルを選択し $z_{ij}^{(\tau+1)}$ とする．
   これに伴い，$\#(z_{ij}^{(\tau+1)}, \boldsymbol{d}_i)$ の値を 1 増やす．

となる．

このステップをすべての $z_{ij} \in Z$ に対して行うことを繰り返す．

3.3.4 まとめ

Q&A ペアごとに異なる混合比率の推定について，簡潔にまとめると次のようになる．

1. 文書集合 D を用意する
2. 3.3.3 節のアルゴリズムにて十分なサンプリングを行う
   超パラメータ $\boldsymbol{\alpha}$ の調整については 4.3.3 節にて検討する．
3. 式 (3.23) により混合比率を計算する

3.4 基本モデル

mixture component としてどのようなモデルを用いるかは重要である．

ひとつの Q&A ペアのモデルを推定するために使えるサンプルは，その質問文と回答文である．質問文・回答文の利用の出発点として，Xue らが論文 [XJC08] で提案した検索モデル（式 (2.47), 式 (2.48)）について検討する．このモデルは，次の 4 つのモデルの混合モデルである．

1. 質問文モデル $P_{\mathrm{ml}}(w|\boldsymbol{q})$
2. 質問文翻訳モデル $\sum_{t\in\boldsymbol{q}} P(w|t)P_{\mathrm{ml}}(t|\boldsymbol{q})$
3. 回答文モデル $P_{\mathrm{ml}}(w|\boldsymbol{a})$
4. 背景分布モデル $P_{\mathrm{ml}}(w|C)$

このモデルを基に，Q&A ペアの情報を最大限に活用し，かつ類似質問という言語現象を効果的に表現するための基本モデルについて検討する．

3.4.1 質問文に基づくモデル

質問文の情報は，類似質問検索において根幹となるものである．まずは，質問文の文書モデル M_q が考えられる．このモデルは最尤推定により推定される．

$$P(w|(\boldsymbol{q},\boldsymbol{a}),M_{\boldsymbol{q}}) = P_{\mathrm{ml}}(w|\boldsymbol{q}) \tag{3.30}$$

しかし，このモデルでは質問文中に現れた語に関する情報しか与えられない．そこで，質問文に現れていないが関係のある語の情報も評価するために，質問文翻訳モデル

$$P(w|(\boldsymbol{q},\boldsymbol{a}),M_{\mathrm{tr}}) = \sum_{t\in\boldsymbol{q}} P(w|t)P_{\mathrm{ml}}(t|\boldsymbol{q}) \tag{3.31}$$

も基本モデルとして考えられる．このモデルは本来,「クエリは適合文書からの翻訳である」という考えから生まれたものであるが，推定量 $P_{\mathrm{ml}}(t|\boldsymbol{q})$ を翻訳確率 $P(w|t)$ により補正していると考えることができる．すなわち，質問文に現れた語から，質問文と関連のある語のモデルを推定していると言える．

Q&A アーカイブを対訳コーパスとみなし，統計的機械翻訳の技術により翻訳確率 $P(w|t)$ を学習する．統計的機械翻訳の技術として，本研究では 2.2.3 節で紹

介した IBM Model 1 を用いた．また，翻訳元と翻訳先の言語が同一であるから，翻訳元の文と翻訳先の文を交換することができる．すなわち，Q&A アーカイブ $\{(\boldsymbol{q},\boldsymbol{a})_1,\ldots,(\boldsymbol{q},\boldsymbol{a})_L\}$ に対して，$\{(\boldsymbol{q},\boldsymbol{a})_1,\ldots,(\boldsymbol{q},\boldsymbol{a})_L,(\boldsymbol{a},\boldsymbol{q})_1,\ldots,(\boldsymbol{a},\boldsymbol{q})_L\}$ を対訳コーパスとして用いた．

3.4.2 回答文に基づくモデル

回答文は，質問文とは異なるが，類似質問検索のために有用な情報を得られると考えられる．
まずは質問文と同様に，回答文の文書モデルが考えられる．このモデルの最尤推定量は

$$P(w|(\boldsymbol{q},\boldsymbol{a}),M_{\boldsymbol{a}}) = P_{\mathrm{ml}}(w|\boldsymbol{a}) \tag{3.32}$$

である．

しかし，このモデルが表現するのは，語の回答文中における出現の情報であり，質問文中に出現することの情報ではない．回答文から，語の質問文中の出現情報を得るために，再び統計的機械翻訳の技術を用いる．質問応答の分野では，質問文から回答文の手がかりを得るために，質問文を翻訳元，回答文を翻訳先として得た翻訳確率を利用する技術がある [SB06]．これを用いると，疑問詞など質問に現れる表現（例：どこ）に対して，回答候補となる表現（例：〜の近く）の確率分布が得られる．逆に，回答文を翻訳元，質問文を翻訳先として学習することで，回答文から質問文に関する情報を得ることができると考えられる．つまり，$\{(\boldsymbol{a},\boldsymbol{q})_1,\ldots,(\boldsymbol{a},\boldsymbol{q})_L\}$ を対訳コーパスとして用いて学習するということである．こうして，新たな独自のモデル

$$P(w|(\boldsymbol{q},\boldsymbol{a}),M_{q|a}) = \sum_{t\in\boldsymbol{a}} P_{Q|A}(w|t)P_{\mathrm{ml}}(t|\boldsymbol{a}) \tag{3.33}$$

を考えることができる．
$P_{Q|A}(w|t)$ は $\{(\boldsymbol{a},\boldsymbol{q})_1,\ldots,(\boldsymbol{a},\boldsymbol{q})_L\}$ を対訳コーパスとして用いて学習した翻訳確率である．このモデルは，回答文に呼応する質問文中の語のモデルということができる．

3.4.3 スムージングのためのモデル

ここまでのモデルのみでは，未知語など，すべてのモデルで出現確率 0 となるような場合，混合モデルでも出現確率 0 となってしまう．従って，そのような語を

1 つでも含むクエリに対するスコアは，強制的に 0 となる．これを防ぐためにはスムージングが必要である．スムージングのために，アーカイブ全体を表すモデル

$$P(w|(\boldsymbol{q},\boldsymbol{a}),M_C) = P(w|M_C) = P_{\mathrm{ml}}(w|C) \tag{3.34}$$

を導入する．

しかしこれが最尤推定量（相対頻度）そのままでは，アーカイブ中に出現しない語について確率 0 としてしまうので，出現頻度に対してスムージングを施しておく．頻度スムージング法としてグッド・チューリング推定法 (Good-Turing estimation) を用いる．これは出現回数 r の補正値として，以下で定義される r^* を用いる．

$$r^* = (r+1)\frac{N_{r+1}}{N_r} \tag{3.35}$$

ここで N_r は，r 回出現した語の異なり数である．

このような補正値を用いた場合，アーカイブ C 中にもともと r 回出現した語 w の出現確率は以下で与えられる．

$$P(w|C) = \begin{cases} \dfrac{r^*}{N} & r > 0 \\ \dfrac{N_1}{N_0 N} & r = 0 \end{cases} \tag{3.36}$$

N は総語数である．

しかし，式 (3.35) の $N_{r+1} = 0$ であると補正ができない．また，r の値が大きい場合は N_r の値が不安定になる（統計的な信頼性が失われる）．そこで実際には，r が小さく，かつ $N_{r+1} > 0$ である場合にのみ式 (3.35) による補正を行い，そうでない場合にはアーカイブから得られた出現回数をそのまま使う．

第 4 章
実験

4.1 共通設定

データセットとして，Yahoo! 知恵袋 [sitc] の研究機関提供用データを用いた．このデータは同サービスベータ版の 2004 年 4 月 1 日から 2005 年 10 月 31 日までの質問と回答の蓄積である．同サービスでは質問に対し 1 つ以上の回答が寄せられ，質問者は寄せられた回答のうちから最も満足なものを必ず 1 つ「ベストアンサー」に選ぶ．データは 3,116,009 件の質問，3,116,008 件のベストアンサー，10,361,777 件のその他の回答からなる．質問及び回答には，質問文・回答文のほかに回答者 ID や参考 URL などの付随情報があるが，本研究では用いない．

実験には同データのうち「インターネット」カテゴリに属するものを用いた．1 つの質問に複数の回答がなされていた場合，1 つ 1 つの回答と当該質問とを対にした別個の Q&A ペアとした．Q&A ペアの総数は 171,816 件であった．文書はすべて MeCab[sitb] により形態素解析を行い，不要語を取り除いた.「インターネット」カテゴリから無作為に 40 件の質問を選び，4.3 節で述べる検索実験に用いるクエリとした．そして同カテゴリからクエリとして選んだ 40 件の質問を含む Q&A ペアを取り除き，検索対象データとした．検索対象データ中の Q&A ペアの総数は 171,759 件となった．変換確率の学習およびモデル M_C の推定は検索対象データにおいて行った．なお変換確率の学習には GIZA++ toolkit[ON03] を利用した．

4.2 基本モデルとサンプリング戦略

まず，採用するモデルと訓練データについて検討した．提案手法のポイントは，

- Q&A ペアを複数の基本モデルの混合により表現する
- 各基本モデルの混合比率はサンプリングに基づいて決定する

という 2 点である．従って基本モデルは，Q&A ペアの性質をよく表現出来るだけでなく，サンプリングの訓練データに過適応しないものでなければならない．また訓練データも，モデルが過学習を起こさないようなものでなければならない．これらの点をクリアするような基本モデルの組と訓練データを検討した．

この実験では，モデルの分布を文書によらず一定とした．従って，完全条件付き確率は式 (3.19) で，混合比率は式 (3.8) により計算される．またハイパーパラメータ $\boldsymbol{\alpha}$ は，すべての場合で $\alpha_1 = \alpha_2 = \cdots = 25$ とした．

基本モデルの組として，$\boldsymbol{M}_1 = \{M_{\boldsymbol{q}}, M_{\mathrm{tr}}, M_{\boldsymbol{a}}, M_C\}$ と $\boldsymbol{M}_2 = \{M_{\boldsymbol{q}}, M_{\mathrm{tr}}, M_{q|a}, M_C\}$ の 2 つを用意した．違いは回答文情報の利用の仕方であり，$\boldsymbol{M}_1$ では回答文の文書モデルをそのまま用いている．一方，$\boldsymbol{M}_2$ では回答文から質問文への翻訳モデルを用いている．

訓練データ D としては，質問文の集合 Q とアーカイブ全体 C が考えられる．$D = Q$ は，混合モデル $P(w|(\boldsymbol{q}, \boldsymbol{a}))$ により文書 $\boldsymbol{q}$ が生成されたとする考えである．一方，$D = C$ の場合，混合モデル $P(w|(\boldsymbol{q}, \boldsymbol{a}))$ により文書 $\boldsymbol{q}$ と $\boldsymbol{a}$ が生成されたという考えである．質問の生成モデルを求めたいという観点からは前者で十分である．しかし，回答文は質問文とは異なるが，話題は共通である．そこで，回答文も訓練データに含めることにより，データ件数を増やし，かつ質問文への過適応を防げる可能性がある．

基本モデルの選び方で 2 通り，訓練データで 2 通り，計 4 通りの条件でギブスサンプリングを行い，混合比率を得た．基本モデルの組が $\boldsymbol{M_1}$ の場合の各モデルの係数を表 4.1 に，$\boldsymbol{M_2}$ の場合を表 4.2 に示す．

まず，訓練データ $D = Q$ であると，モデルの組に関わらず $M_{\boldsymbol{q}}$ の重みが 1 となり，他のモデルの重みが 0 となってしまうことがわかった．モデル $M_{\boldsymbol{q}}$ は質問文の最尤推定による文書モデルであり，質問文そのものを最も良く表現している．従って，質問文そのものの各語を生成したモデルはほぼ間違いなく常に $M_{\boldsymbol{q}}$ と推定されるのは当然である．しかし，質問文の文書モデルを用いた類似質問検索の性能

表 4.1: モデル組 $\boldsymbol{M}_1$ における各モデルの重み

training data D	Weight for each model			
	$M_{\boldsymbol{q}}$	M_{tr}	$M_{\boldsymbol{a}}$	M_C
Q	1.0000	0.0000	0.0000	0.0000
C	0.5204	0.0000	0.4796	0.0000

表 4.2: モデル組 $\boldsymbol{M}_2$ における各モデルの重み

training data D	Weight for each model			
	$M_{\boldsymbol{q}}$	M_{tr}	$M_{q\|a}$	M_C
Q	1.0000	0.0000	0.0000	0.0000
C	0.5155	0.1039	0.3795	0.0012

が著しく劣ることは既存研究にて指摘されている．すなわち，これは訓練データにおいて過学習が起きていると解釈できる．

次に，訓練データ $D = C$ でも基本モデルの組が $\boldsymbol{M}_1$ の場合は $M_{\boldsymbol{q}}$ と $M_{\boldsymbol{a}}$ で重みを二分してしまうことがわかる．モデル $M_{\boldsymbol{a}}$ は最尤推定による回答文の文書モデルであり，回答文そのものをよく表現している．従って，この結果は，質問文の各語を生成したモデルは $M_{\boldsymbol{q}}$，回答文の各語を生成したのは $M_{\boldsymbol{a}}$ と推定されたためであると考えられる．このような重みを用いた場合，質問文と回答文それぞれの文書モデルの finite mixture による検索モデルとなる．しかしながら，その性能は translation-based language model に劣ることが示されている．よってこれもまた過学習であると言える．

過学習が起きていないのは，基本モデル組として $\boldsymbol{M}_2$ ，訓練データとして C を選んだ場合である．よってここからは基本モデル組として $\boldsymbol{M}_2$，訓練データとして C を用いて実験を行っていくこととした．

また表 4.3 はモデル組 $\boldsymbol{M}_2$，訓練データ $D = C$ の場合の，各モデルの重みの収束の様子を表している．run=0 すなわち初期状態では，モデルの初期値は一様にしてあるので，重みはほぼ 4 等分となっている．サンプリングを繰り返すにつれてある値に収束していくのがわかる．

表 4.3: 混合重みの収束の様子．ただしモデル組は $\boldsymbol{M}_2$，訓練データ $D = C$ である

	Weight for each model			
run	$M_{\boldsymbol{q}}$	M_{tr}	$M_{q\|a}$	M_C
0	0.2501	0.2501	0.2497	0.2502
1	0.4721	0.1797	0.3178	0.0304
2	0.5090	0.1333	0.3536	0.0041
3	0.5137	0.1154	0.3694	0.0015
4	0.5147	0.1084	0.3756	0.0012
5	0.5153	0.1054	0.3781	0.0012
6	0.5154	0.1045	0.3789	0.0012

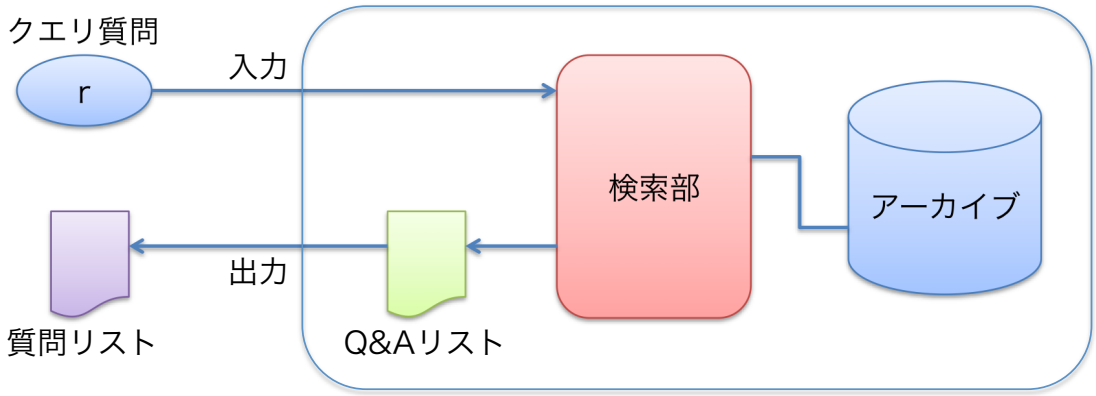

図 4.1: 検索システムの概略

4.3 検索実験

提案手法のパラメータ調整・性能の比較を行うために，検索システムを構築して実験を行った．

実際の Q&A 検索システムにおける検索結果は，まず Q&A ペアの質問文だけが適合度順に並べられ，ユーザがそのうち 1 つを選ぶと対応する回答文が表示されるという形態が便利である．回答文もはじめから表示すると，1 つの質問に対し多数の回答がなされている場合に，画面が 1 つの Q&A スレッドで埋め尽くされてしまい不便である．通常，質問に対する回答の適合性は保証されていると考えられる [JCLP06] ので，検索システムの性能は，そのシステムが返す，クエリに適合する質問の順位で評価することができる．そこで，適合性の判定は質問文に基づいて行うこととした．順位付けは Q&A ペアに対してなされるが，同じ質問を含む Q&A ペアのうちから最も高順位のものを選ぶことにより，Q&A ペアの順位を質問文の順位に変換することができる．以降の実験では，順位付けアルゴリズムは，まず Q&A ペアの順位付けを行い，ついで質問文の順位に変換する．検索システムを図で表現すると図 4.1 のようになる．

4.3.1 テストコレクションと評価指標

4.1 節で述べたように，ランダムに選んだ 40 件の質問をクエリとした．各手法の検索結果の上位 20 件を pooling した．pooling とは，複数の検索手法による，同一クエリに対するそれぞれの検索結果の上位 x 件を合併し，それらに対してのみ適合判定を行う方法である [岸田 24]．pooling により得られた適合候補質問が，ク

表 4.4: クエリ質問と適合 Q&A の例

クエリ質問	Yahoo!等の検索欄をダブルクリックすると過去に検索した履歴が出て きます。消せないのでしょうか。 会社の LAN に繋がったパソコン（Windows98）で Yahoo!、goo、Google 等の検索欄をダブルクリックすると過去に検索した履歴が出てきます。 消せないのでしょうか。 よろしくお願いします。
適合質問	検索するときの四角い箱の中で左クリックすると以前検索した言葉がでてきますが、 その履歴は消せないんでしょうか？
適合質問への回答	出てきた言葉の上にカーソルをもっていて、Delete を押す (windows me)

エリ質問と類似しているかどうか人手で判定し，正解セットを作成した．正解判定は 1 人で行った．クエリ質問と適合 Q&A の例を表 4.4 に示す．正解判定の結果，適合質問が 1 件も見つからなかったときは，そのクエリを無効とした．結果として，有効なクエリは 32 件となった．計 571 件の正解が得られた．なお pooling には，4.3.2, 4.3.4 両節にて扱うすべての手法の検索結果を用いたので，正解セットは両節の実験で共通である．

pooling に用いた手法は

- 言語モデル（式 (2.30), 式 (2.34)）
- Okapi/BM25[SJWR00]
- translation-based language model + query likelihood （式 (2.47)）
- Method 1
- Method 2-1
- Method 2-2

である．Method 1, 2-1, 2-2 は提案手法である．Method 1 については 4.3.2 節で，2-1 および 2-2 については 4.3.3 節で述べる．また，クエリ $\boldsymbol{r}$ に対する文書 $\boldsymbol{d}$ の

Okapi/BM25 のスコアは以下のように与えられる．

$$\mathrm{BM25}(\boldsymbol{d}, \boldsymbol{r}) = \sum_{w \in \boldsymbol{r}} \mathrm{IDF}_{\mathrm{BM25}}(w)\mathrm{TF}_{\mathrm{BM25}}(w, \boldsymbol{d})\frac{(k_3+1)\mathrm{qtf}(w)}{k_3 + \mathrm{qtf}(w)} \tag{4.1}$$

$$\mathrm{IDF}_{\mathrm{BM25}}(w) = \log \frac{L - \mathrm{df}(w) + 0.5}{\mathrm{df}(w) + 0.5} \tag{4.2}$$

$$\mathrm{TF}_{\mathrm{BM25}}(w, \boldsymbol{d}) = \frac{\#(w, \boldsymbol{d}) \cdot (k_1 + 1)}{\#(t, \boldsymbol{d}) + k_1(1 - b + b\dfrac{|\boldsymbol{d}|}{\mathrm{avgdl}})} \tag{4.3}$$

ここで qtf(w) はクエリ $\boldsymbol{r}$ 中に語 w が出現した頻度，df(w) は語 w の出現した文書の数 (document frequency), avgdl は平均の文書長である．また b, k_1, k_3 はパラメータである．

なお言語モデル，Okapi/BM25 の 2 つの手法は，質問文と回答文を結合したものを検索対象としている．これは，これらの手法において質問文のみを用いるより，質問文と回答文を結合して用いる方が性能が向上するためである．また既存手法及び Method 1 においてはパラメータチューニングを行っているので，その際の結果も pooling に含めている．

Okapi/BM25 の実装には lemur toolkit[sita] を利用した．

1 つのクエリによる検索の評価指標として，上位 10 件の適合率 (P@10) と平均精度 (average precision) v の 2 つを用いた．

上位 i 件目の文書が適合文書かそうでないかを表す変数を x_i とし，適合ならば $x_i = 1$，不適合ならば $x_i = 0$ とする．

P@10 は次の式で与えられる．

$$\mathrm{P@10} = \frac{\sum_{i=1}^{10} x_i}{10} \tag{4.4}$$

平均精度 v は次の式で与えられる．

$$v = \frac{1}{\sum_i x_i} \sum_i \left[\frac{x_i}{i} \left(1 + \sum_{k=1}^{i-1} x_k \right) \right] \tag{4.5}$$

平均精度は簡単にいえば,「各適合文書が検索された時点での精度の平均」を意味する [BV00].

さらに，テストコレクション全体での評価指標として，各クエリごとの，P@10 の平均 (mean P@10) と 平均精度の平均 (mean average precision: MAP) を用いる．

表 4.5: $\boldsymbol{\alpha}$ のオーダー変化に対する各モデルの重み

	Weight for each model			
α_k	$M_{\boldsymbol{q}}$	M_{tr}	$M_{q\|a}$	M_C
15000	0.5123	0.1074	0.3758	0.0044
150000	0.4860	0.1321	0.3526	0.0293
1500000	0.3648	0.2107	0.2855	0.1390
15000000	0.2678	0.2458	0.2549	0.2314

4.3.2 ハイパーパラメータ α の調整

4.2 節で過学習が起きないような訓練データ・モデル組を検討した．しかし，訓練データは質問という言語現象のサンプルに過ぎないので，なお過学習が起きる可能性がある．この場合，モデルのサンプリング・分布推定における超パラメータ $\boldsymbol{\alpha}$ が，過学習を抑制する役割を果たす．$\boldsymbol{\alpha}$ の値を決定するにあたり，考えるべきことは次の 2 点である．

- どの程度のオーダーが望ましいか
- どのような傾向をもたせるのが望ましいか

前者は $\boldsymbol{\alpha}$ の寄与の大きさ，後者は寄与の性質を決定する．後者は，事前に各モデルの出現傾向を推定するのは難しいことから，$\alpha_1 = \alpha_2 = \cdots = \alpha_K$ としておくのが良いと考えられる．一方前者は，ひとまず訓練データのサイズ（総語数）に対する割合を検討することができる．

混合重み一定の場合について，最適な $\boldsymbol{\alpha}$ のオーダーを検証する実験を行った．4.2 節と同様のサンプリング・混合重み決定の後，式 (3.1) および式 (3.1) により適合度を計算した．訓練データの総単語数がおよそ 6,000,000 語であったことから，

$$\sum_k \alpha_k = 60000, 600000, 6000000, 60000000$$

と変えて検索を行い，最も良い性能を得るものを探した．なお，$\alpha_1 = \alpha_2 = \cdots = \alpha_K$ とした．$K = 4$ であるから，$\alpha_k = 15000, 150000, 1500000, 15000000$ と変化する．$\boldsymbol{\alpha}$ のオーダー変化に対する混合重みの変化を表 4.5 に，検索性能の変化を表 4.6 に示す．

表 4.6: $\boldsymbol{\alpha}$ のオーダー変化に対する P@10 と MAP の値

α_k	mean P@10	MAP
15000	0.3719	0.3766
150000	0.3781	0.4041
1500000	0.3813	0.4067
15000000	0.3781	0.4036

表 4.5 からは，$\boldsymbol{\alpha}$ のオーダーを大きくするにつれ，モデルの重みが等しくなっていくのが見て取れる．特に $\alpha_k = 15000000$ すなわち $\sum_k \alpha_k$ が訓練データサイズの 10 倍ほどになるとほぼ 4 等分となっている．

一方表 4.6 からは，P@10, MAP ともに $\alpha_k = 1500000$ のとき最も良いことがわかる．このとき $\sum_k \alpha_k$ は訓練データサイズとほぼ同等である．このように大きなサイズが必要となったということは，訓練データは未だサンプルとして不十分であるということを示している．訓練データについては検討の余地がある．

現在の混合モデル・訓練データでは，超パラメータのオーダーと訓練データサイズは同等程度が良いと考えられる．今後，混合重み一定で $\alpha_k = 1500000$ とする手法を Method 1 とする．

4.3.3 混合重みを Q&A ごとに変える

ここまでの知見を基に，混合重みを Q&A ごとに変える手法について検討する．サンプリングのアルゴリズムは 3.3.3 節で述べたとおりである．そこで，ディリクレ事前分布のパラメータ $\boldsymbol{\alpha}$ の決定について考える．オーダーについては，4.3.2 節と同様，訓練データサイズと同等にする．訓練データの平均文書長はおよそ 35 であったため，$\sum_k \alpha_k = 35$ とする．一方，$\boldsymbol{\alpha}$ の傾向については，2 通りの方策が考えられる．1 つは，4.3.2 節と同様，$\alpha_1 = \alpha_2 = \cdots = \alpha_K$ とすることである．もう 1 つは，Method 1 の混合重みの定数倍とすることである．すると，あたかも混合重みのディリクレスムージングのように振舞う．前者を Method 2-1, 後者を Method 2-2 とする．Method 2-1 の混合重みの例を表 4.7 に，Method 2-2 の場合を表 4.8 に示す．

Method 2-1, 2-2 共に Q&A ペアごとに混合重みが変化していることがわかる．また，表 4.5, 表 4.7, 表 4.8 と比較すると，同じ Q&A ペアに対しても混合重みが異

表 4.7: Method 2-1 の混合重みの例

Q&A pair	Weight for each model			
	$M_{\boldsymbol{q}}$	M_{tr}	$M_{q\|a}$	M_C
$(\boldsymbol{q},\boldsymbol{a})_1$	0.2448	0.2284	0.3672	0.1793
$(\boldsymbol{q},\boldsymbol{a})_2$	0.4457	0.1786	0.2643	0.1364
$(\boldsymbol{q},\boldsymbol{a})_3$	0.3760	0.1979	0.2573	0.1823

表 4.8: Method 2-2 の混合重みの例

Q&A pair	Weight for each model			
	$M_{\boldsymbol{q}}$	M_{tr}	$M_{q\|a}$	M_C
$(\boldsymbol{q},\boldsymbol{a})_1$	0.3150	0.1961	0.4042	0.1045
$(\boldsymbol{q},\boldsymbol{a})_2$	0.5267	0.1361	0.2899	0.0723
$(\boldsymbol{q},\boldsymbol{a})_3$	0.4702	0.1703	0.2717	0.1013

なることがわかる．

4.3.4 手法の性能比較

提案手法の性能を検証するために，さらに検索実験を行った．まず提案手法は，4.3.2 節で用いた Method 1, 4.3.3 節で述べた Method 2-1, 2-2 である．比較対象は，まず最新の文書検索モデルの代表である言語モデル (LM) および Okapi/BM25 (BM25) をベースラインとした．また，類似質問検索の既存手法として Xue らの検索モデル (transLM+QL) を比較対象とした．
結果を表 4.9 に示す．

まず，ベースラインである言語モデルおよび Okapi/BM25 と，その他の手法の間に，どちらの評価指標でも大きな差があることがわかる．すなわち，類似質問検索に特化した手法の有効性が示されている．次に，そのような特化した手法である，表の下部 4 手法内での比較を行う．mean P@10 においては，Method 1 が最も優れていて，次に Method 2-2 が優れている．Xue らの手法と Method 2-1 は同じスコアを示している．一方 MAP においては，最も優れているのは同じく Method 1 であり，その後 Method 2-1, 2-2, Xue らの手法と続く．これらのことから，提案手法のいずれも Xue らの既存手法に対し優れていることがわかる．特に Method 1 は

表 4.9: 手法の性能比較

	mean P@10	MAP
LM	0.2686	0.2809
BM25	0.2844	0.3107
transLM+QL	0.3625	0.3629
Method 1	0.3813	0.4067
Method 2-1	0.3625	0.4056
Method 2-2	0.3688	0.3965

表 4.10: 有意性の検定結果．a は MAP, p は mean P@10 において有意水準 95% で有意差が得られたことを示す

significance	LM	BM25	transLM+QL
Method 1	a, p	a, p	a
Method 2-1	a, p	a, p	a
Method 2-2	a, p	a, p	

いずれの評価指標においても最も優れていると言える．性能の差であるが， mean P@10 では Method 2-1, Method 2-2, Xue らの手法がほぼ同等であり，Method 1 はそれらを 2% ほど上回っている．MAP では， 提案手法のいずれも Xue らの手法を 4% ほど上回っている．

次に有意性を調べるため検定を行った．同一のクエリにおける評価指標を比較するので，一連の値は対応のあるデータ，すなわち対標本 (paired data) と考えられる [Rob90]．そこで，2 つのデータの差をとり，その平均が 0 であるという仮説を棄却することで有意性が示せる．また，サンプル数は 32 と小さいので，t 検定を用いる．対応のある両側 t 検定では，2 つの手法の一連の評価値（P@10 または平均精度）を対標本と見なして検定する．検定統計量 t は次のようになる．

$$t = \frac{\bar{d}}{\frac{s_d}{\sqrt{n}}} \tag{4.6}$$

ここで $\bar{d}$ は各データの差の平均, s_d は差の標準偏差，n はデータ数で $n = 32$ である．

有意性の検定結果を表 4.10 に示す．a は MAP, p は mean P@10 において有意水

準 95% で有意差が得られたことを表す．表 4.10 から，提案手法のいずれも，ベースライン手法に対しては，mean P@10, MAP 共に有意に優れていることがわかる．また Xue らの既存手法に対しては，Method 1 および 2-1 は，MAP においては有意に優れていることがわかる．

第 5 章
結論

5.1 まとめ

Q&A アーカイブは，豊富な情報を持つ知識源である．Q&A アーカイブにおける類似質問検索は，通常の文書検索と異なる．類似質問検索特有の問題として，word overlap が少ないということが挙げられる．

既存研究ではこれに対し複数の確率的モデルを手動で組み合わせることで良い性能を得ている．筆者はこれを拡張し，Q&A を複数のモデルの混合モデルとして表現することを提案した．あわせて，その混合比率の推定手法を提案した．さらに，基本モデルとして既存手法より効果的なものを提案した．

混合比率の扱いについて，アーカイブ全体で一定とする方法と，Q&A ごとに異なるとする方法の 2 つを提案した．そのどちらについても，混合比率を推定する手法を述べた．ギブスサンプリングを行い，予測分布として推定する．

基本モデルとして，既存手法にて提案されているものに加え，回答文から質問文の情報を得るモデルを提案した．

実験を通して，提案手法の有効性を検証した．混合比率推定のための訓練データと，基本モデルの組み合わせを模索し，効果的な学習ができる訓練データと基本モデルの組み合わせを発見した．合わせて，超パラメータの最適なオーダーについて 1 つの知見を得た．

性能比較においては，混合比率が Q&A ごとに異なるより，アーカイブ全体で一定とする方が性能が良いことを明らかにした．さらに最新の文書検索手法及び類似質問検索の既存手法との比較を行った．その結果，提案手法が優れていることを示した．

5.2 今後の展望

混合モデルの有効性を検証するためには，以下のような実験が必要だと考えている．

- 他の確率的モデルを基本モデルとして採用する
  合わせて，良い混合比率を推定できるかどうか検証する
- 他の Q&A アーカイブにおいても実験を行う

実用的な類似質問検索システムのためには，回答の質をある程度保証するなど，検索精度以外の観点も盛り込む必要があると考えられる．

謝辞

本研究および修士課程での学生生活において，様々な面で協力いただき，支えてくださった方々に対し，ここに感謝の意を表したいと思います．

指導教員の安達淳教授には，本研究を進め本論文を執筆するにあたり，懇切で熱心なご指導を賜りました．ミーティングでは，研究に限らず多方面にわたる様々な助言をいただき，幅広いご指導をいただいたことに，心より感謝申し上げます．

国立情報学研究所の高須淳宏教授には，研究の方向性から細部にいたるまで重要な助言と指導をいただきました．大変感謝いたしております．

安達研究室の方々には，日常の様々な面でお世話になりました．博士課程 2 年の倉沢央さん，博士課程 1 年の Chu Yimin さん，修士課程 1 年の木村光樹さん，修士課程 1 年の渡辺健太郎さんには，研究生活において様々な助言をいただきました．また，皆様のおかげで楽しい研究生活を送ることができました．大変感謝申し上げます．

最後に，学生生活において，支え協力していただいた両親，家族をはじめとする皆様に感謝申し上げます．

本研究の実施にあたっては，ヤフー株式会社が国立情報学研究所に提供した Yahoo! 知恵袋データを利用しました．

参考文献

[AA84] S.G.E.M. AN and D.G.E.M. AN. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Machine Intell*, Vol. 6, pp. 721–741, 1984.

[BDPDPM93] P.F. Brown, V.J. Della Pietra, S.A. Della Pietra, and R.L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, Vol. 19, No. 2, pp. 263–311, 1993.

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[BL99] Adam L. Berger and John D. Lafferty. Information retrieval as statistical translation. In *SIGIR*, pp. 222–229, 1999.

[BNJ03] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, Vol. 3, pp. 993–1022, 2003.

[BS09] Roger E. Bohn and James E. Short. How much information? 2009 report on american consumers, dec 2009.

[BV00] C. Buckley and E.M. Voorhees. Evaluating evaluation measure stability. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 33–40. ACM New York, NY, USA, 2000.

[FEL98] C. FELLBAUM. *WordNet an Electronic Lexical Database*, 1998.

[HMK09] F. Maxwell Harper, Daniel Moy, and Joseph A. Konstan. Facts or friends?: distinguishing informational and conversational questions in social q&a sites. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, pp. 759–768, New York, NY, USA, 2009. ACM.

[HRRK08] F. Maxwell Harper, Daphne R. Raban, Sheizaf Rafaeli, and Joseph A. Konstan. Predictors of answer quality in online q&a sites. In *CHI*, pp. 865–874, 2008.

[JCL05] Jiwoon Jeon, W. Bruce Croft, and Joon Ho Lee. Finding similar questions in large question and answer archives. In *CIKM*, pp. 84–90, 2005.

[JCLP06] Jiwoon Jeon, W. Bruce Croft, Joon Ho Lee, and Soyeon Park. A framework to predict the quality of answers with non-textual features. In *SIGIR*, pp. 228–235, 2006.

[MP00] G.J. McLachlan and D. Peel. *Finite mixture models*. Wiley-Interscience, 2000.

[MSI08] Tatsunori Mori, Mitsuru Sato, and Madoka Ishioroshi. Answering any class of japanese non-factoid question by using the web and example q&a pairs from a social q&a website. In *Web Intelligence*, pp. 59–65, 2008.

[ON03] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, Vol. 29, No. 1, pp. 19–51, 2003.

[PC98] Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *SIGIR*, pp. 275–281, 1998.

[Rob90] SE Robertson. On Sample Sizes for Non-Matched-Pair IR Experiments. *Information Processing and Management*, Vol. 26, No. 6, pp. 739–53, 1990.

[SB06] Radu Soricut and Eric Brill. Automatic question answering using the web: Beyond the factoid. *Inf. Retr.*, Vol. 9, No. 2, pp. 191–206, 2006.

[sita] The lemur toolkit for language modeling and information retrieval. *http://www.lemurproject.org/*.

[sitb] Mecab yet another part-of-speech and morphologicalanalyzer. *http://mecab.sourceforge.net*.

[sitc] yahoo 知恵袋. *http://chiebukuro.yahoo.co.jp/*.

[SJWR00] K. Sparck Jones, S. Walker, and SE Robertson. A probabilistic model of information retrieval: development and comparative experiments. *Information Processing and Management*, Vol. 36, No. 6, pp. 809–840, 2000.

[Tay67] R.S. Taylor. QUESTION-NEGOTIATION AN INFORMATION-SEEKING IN LIBRARIES., 1967.

[WMC09] Kai Wang, Zhaoyan Ming, and Tat-Seng Chua. A syntactic tree matching approach to finding similar questions in community-based qa services. In *SIGIR*, pp. 187–194, 2009.

[XJC08] Xiaobing Xue, Jiwoon Jeon, and W. Bruce Croft. Retrieval models for question and answer archives. In *SIGIR*, pp. 475–482, 2008.

[岸田 24] 岸田和明. 検索実験における評価指標としての mean average precision の性質. 情報処理学会研究報告. 情報学基礎研究会報告, Vol. 2001, No. 74, pp. 97–104, 20010724.

[北 99] 北研二. 確率的言語モデル. 東京大学出版会, 1999.

発表文献

[1] 高橋輝, 高須淳宏, 安達淳, “コミュニティベース Q&A からの類似質問検索手法”, **情報処理学会全国大会**, 6ZC-2, 2010. (発表予定)