Bring you to the past: Automatic Generation of Topically Relevant Event Chronicles

Authors:

Tao Ge, Wenzhe Pei, Heng Ji, Sujian Li, Baobao Chang, Zhifang Sui (Peking University)

Conference:

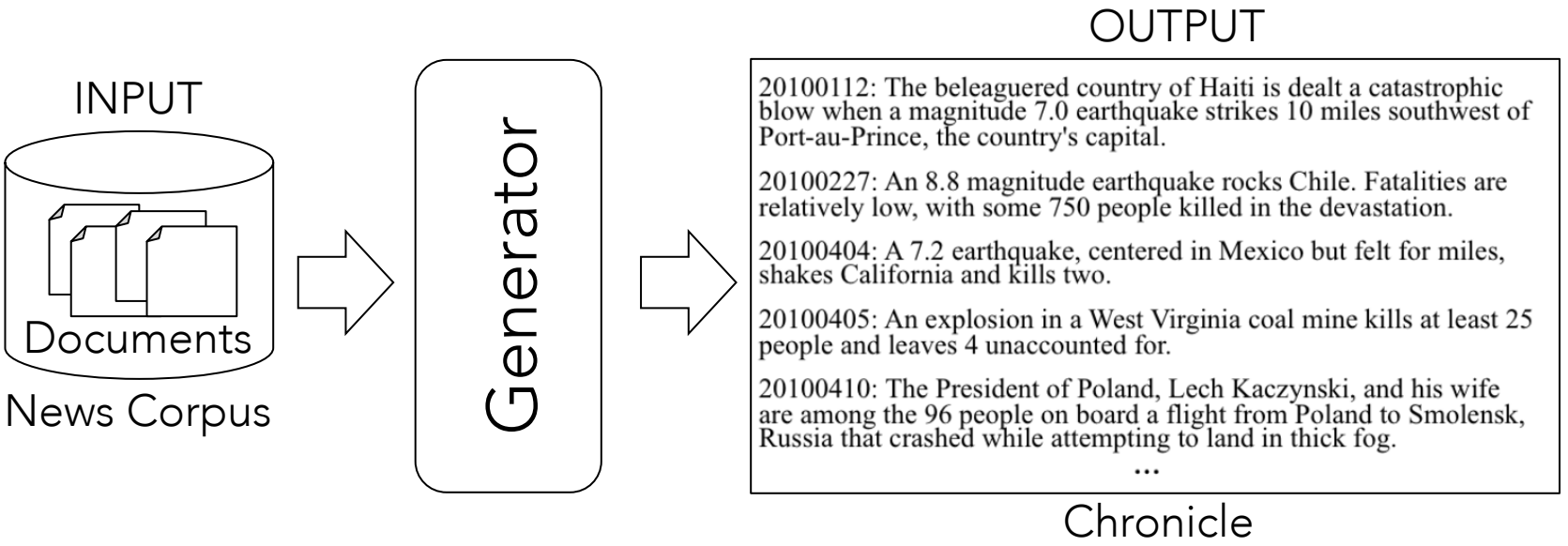
ACL2015

Expositor:

Kento Watanabe (Tohoku University)

Task: Chronicle Generation

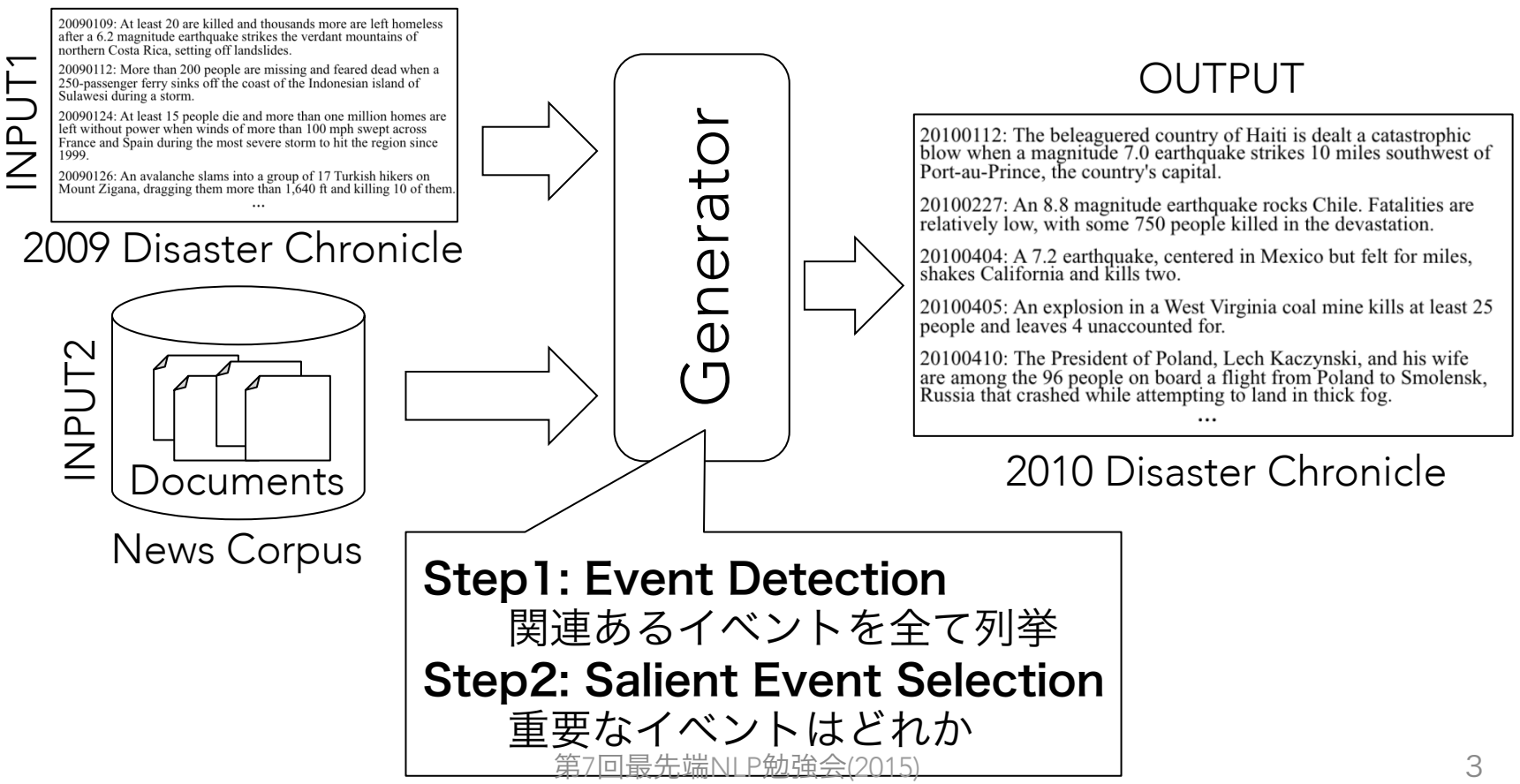
- **Chronicle**：重要なeventの詳細を時系列ごとに記す
- 人手での作成にはコストが大きい



- DocumentsからChronicleに必要なイベントを探す
  - Multi Doc Summarization，Timeline Generationに近いタスク
- トピックによってChronicleは異なる
  - 災害Chronicleなら地震イベントのみを使う．野球イベントは使わない

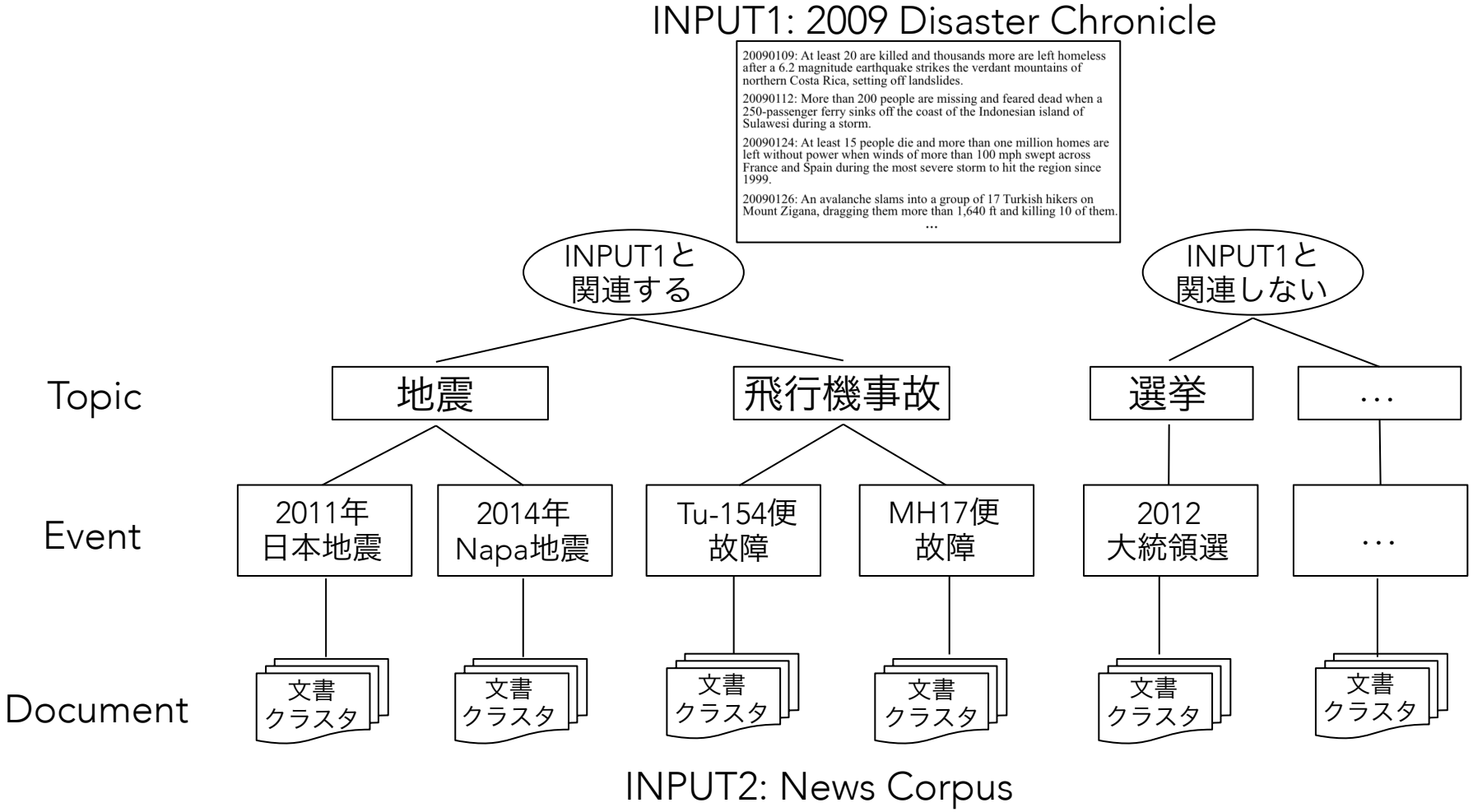
Topically Relevant Event Chronicle Generation

- 2009年のChronicleとトピックに関連があるEventを Documentsから探す



STEP1：EVENT DETECTION

このイベントは必要？不必要？



目標：この階層的構造を確率モデル化し推定する

このイベントは必要？不必要？

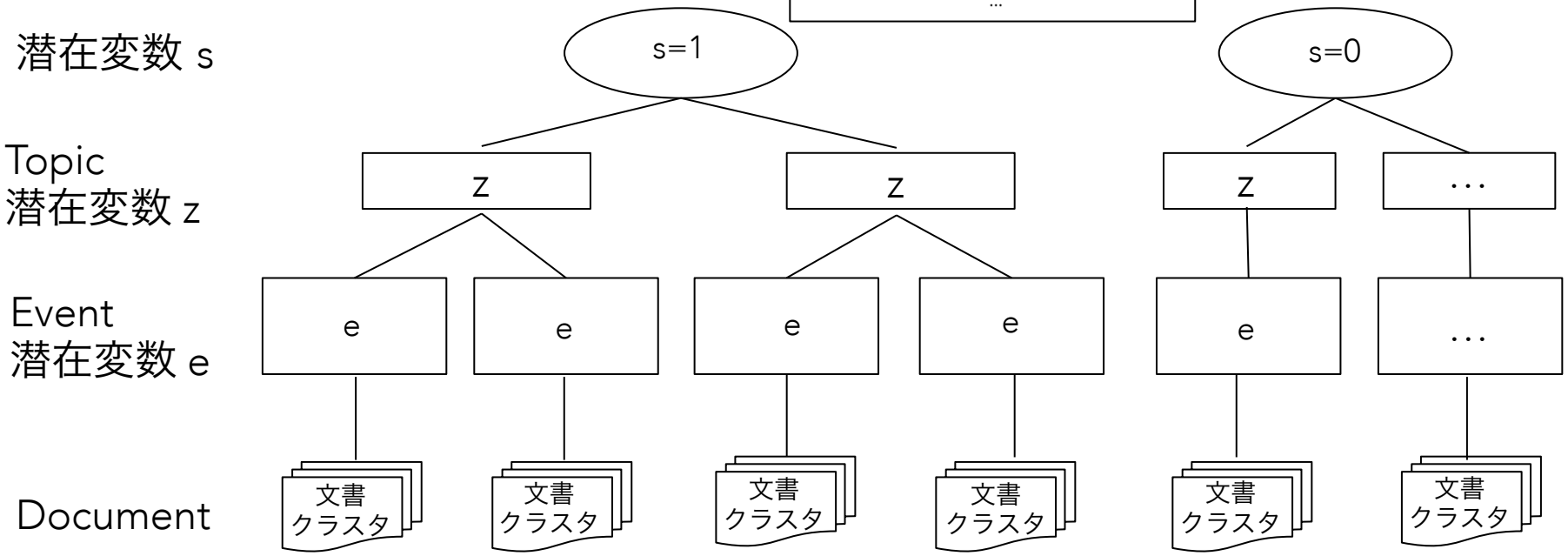
INPUT1: 2009 Disaster Chronicle

20090109: At least 20 are killed and thousands more are left homeless after a 6.2 magnitude earthquake strikes the verdant mountains of northern Costa Rica, setting off landslides.

20090112: More than 200 people are missing and feared dead when a 250-passenger ferry sinks off the coast of the Indonesian island of Sulawesi during a storm.

20090124: At least 15 people die and more than one million homes are left without power when winds of more than 100 mph swept across France and Spain during the most severe storm to hit the region since 1999.

20090126: An avalanche slams into a group of 17 Turkish hikers on Mount Zigana, dragging them more than 1,640 ft and killing 10 of them.
...

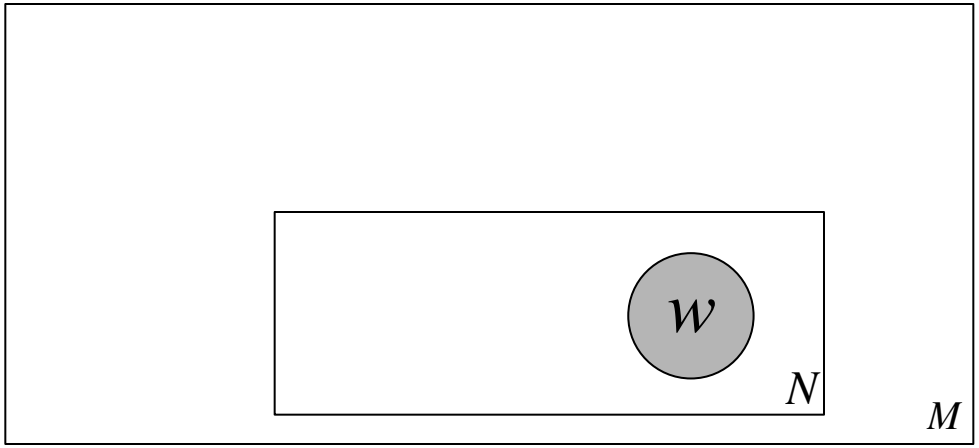


INPUT2: News Corpus

目標：P(s|e) = P(s) * P(s|z) * P(e|z) / P(e) の計算
よくある階層的ベイジアンモデル

Hierarchical Bayesian Model

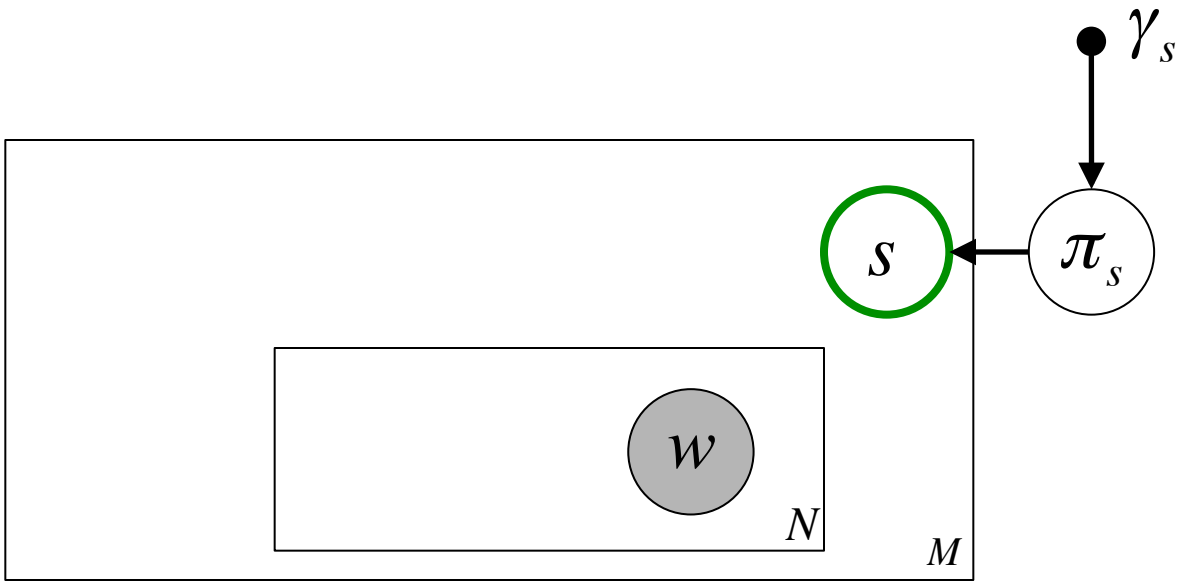
- m番目の文書d(N個の単語wを持つ)において



Hierarchical Bayesian Model

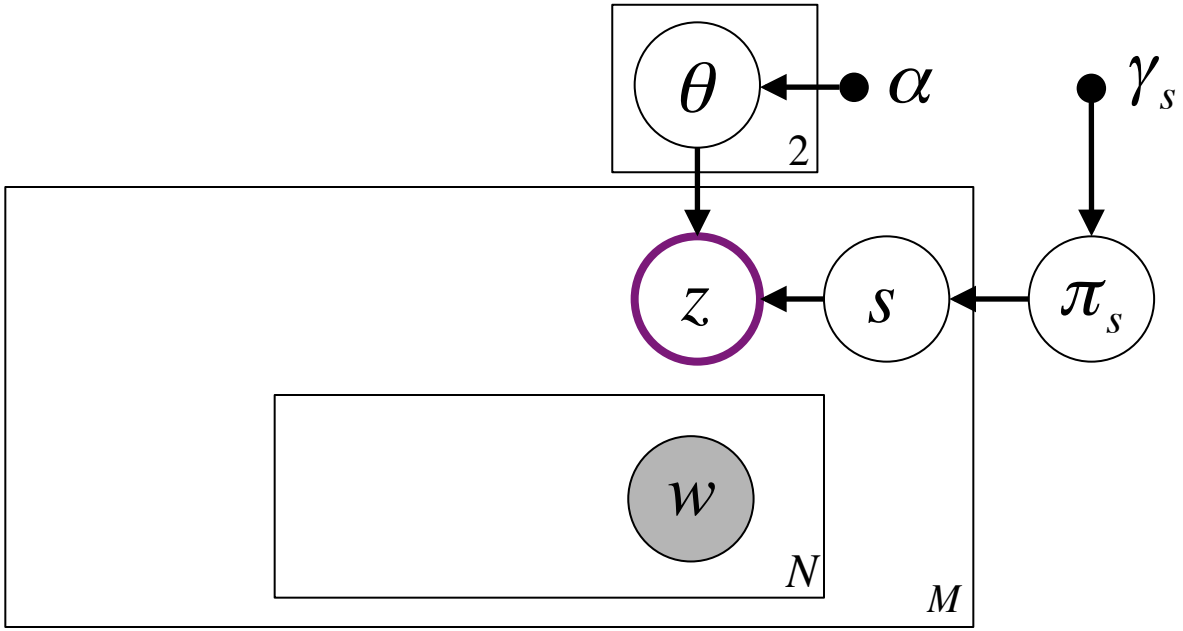
- 文書が**イベントと関連がある(s=1)かないか(s=0)**ベルヌーイ分布により生成

$$s \sim Bernoulli(\pi_s) \qquad \pi_s \sim Beta(\gamma_s)$$



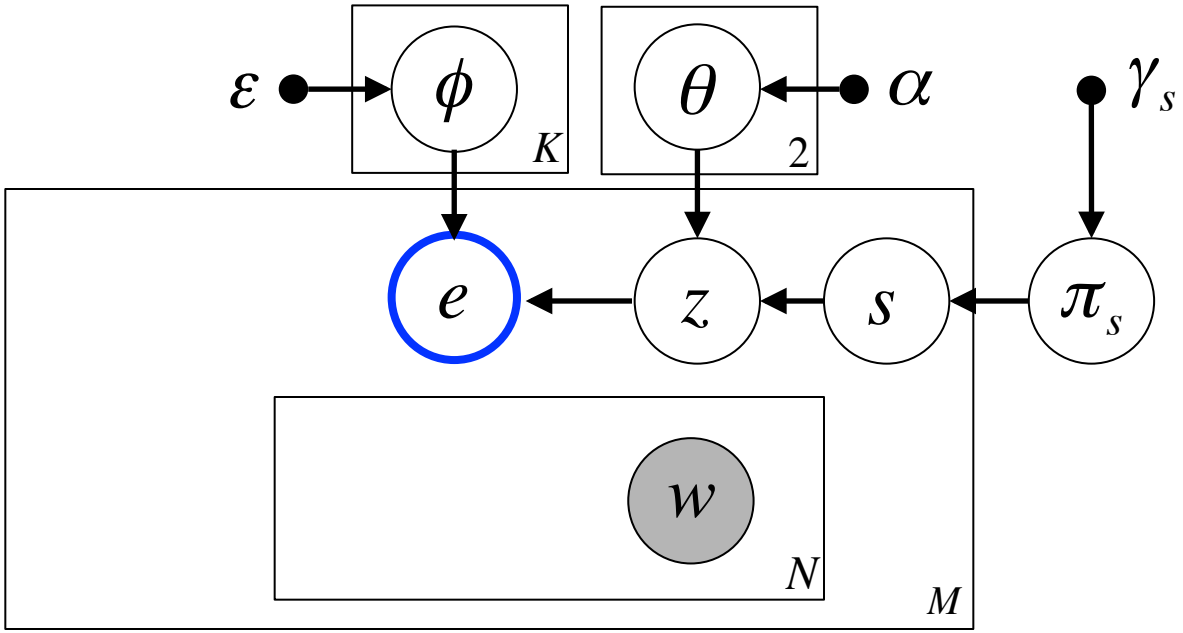
Hierarchical Bayesian Model

- **トピックz**は多項分布によって生成 $z \sim Multi(\theta^{(s)}) \quad \theta^{(s)} \sim Dir(\alpha)$



Hierarchical Bayesian Model

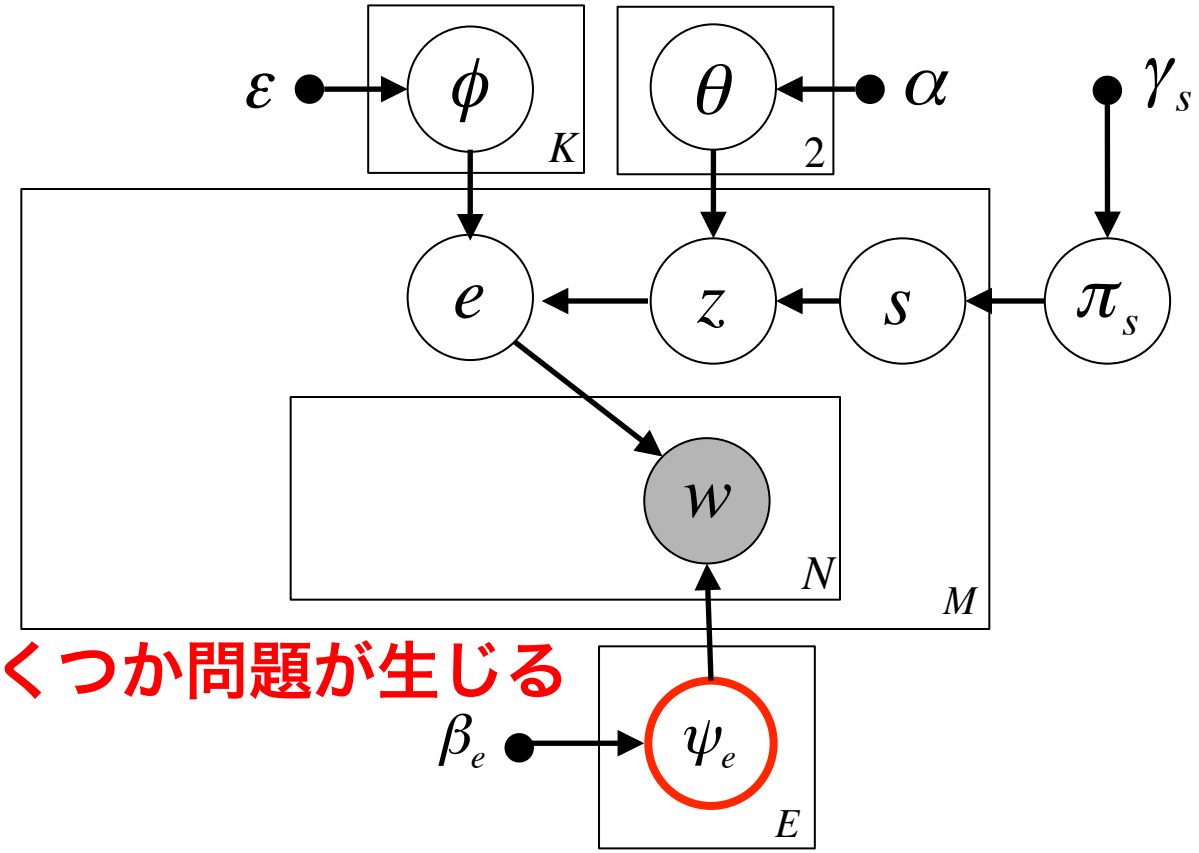
- イベントeは多項分布によって生成 $e \sim Multi(\phi^{(z)})\ \phi^{(z)} \sim Dir(\varepsilon)$



Hierarchical Bayesian Model

- 普通の階層モデルならイベントeの単語生成分布より単語wが生成される

$$w \sim \psi_e^{(e)} \qquad \psi_e^{(e)} \sim Dir(\beta_e)$$



このままだといくつか問題が生じる

イベントを正しく区別するためには

問題1：時間情報を考慮する必要がある

(d_1) A magnitude-6.1 earthquake in southern China's Yunnan province destroyed thousands of home on Sunday, killing at least 367 people. (Aug 2014)

(d_2) A massive earthquake devastated Sichuan province in southwest China, leaving five million people homeless and taking some 87,000 lives. (May 2008)

どちらも中国の地震の文書

日付が違うことからこれらは別のイベントであると分かる

問題2：トピック特有の情報とイベント特有の情報を区別

(d_1) A magnitude-6.1 earthquake in southern China's Yunnan province destroyed thousands of home on Sunday, killing at least 367 people. (Aug 2014)

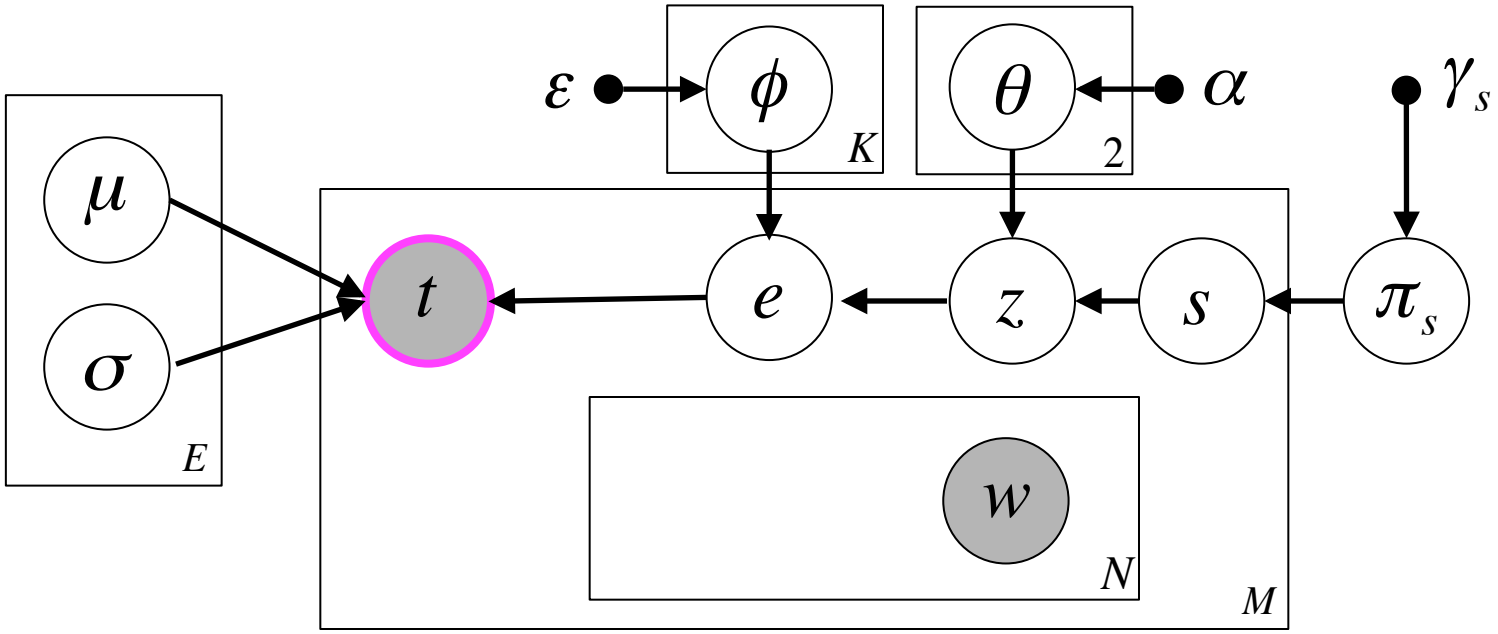
(d_3) A 3.4-magnitude earthquake occurred in and around the city of Napa, California on Sunday, killing one person and injuring about 200. (Aug 2014)

どちらもAug 2014の文書
どちらも地震に関する文書

イベントの場所が違うことから別のイベントであると分かる

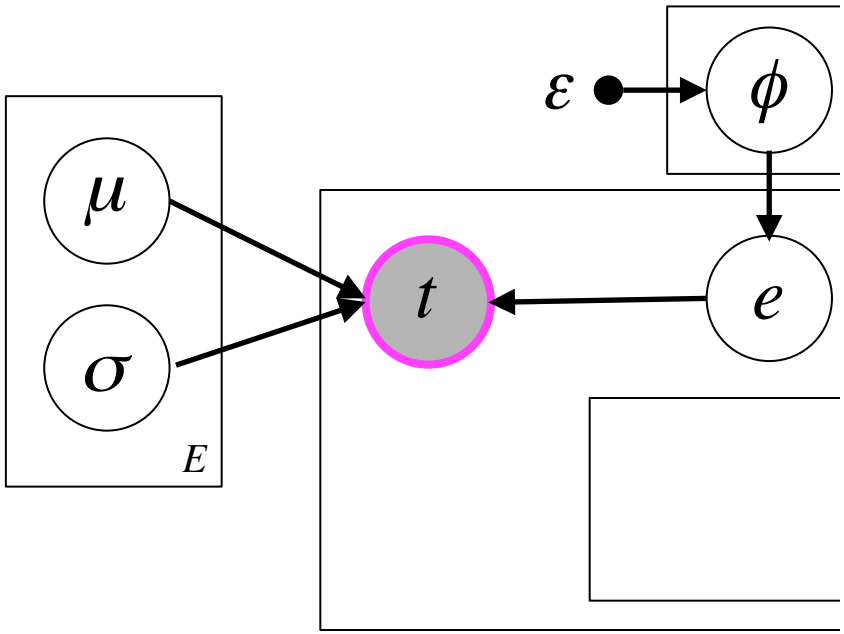
Time-aware Hierarchical Bayesian Model
時間情報の導入

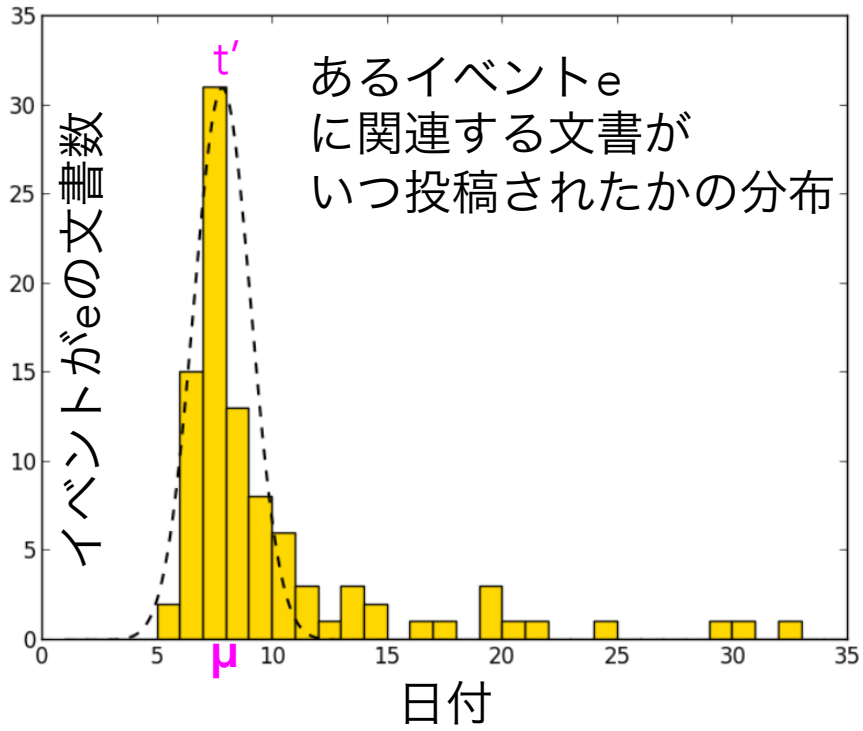
- タイムスタンプtは**正規分布**によって生成 $t' \sim Gaussian(\mu_e, \sigma_e)$ $t \leftarrow \lfloor t' \rfloor$



Time-aware Hierarchical Bayesian Model
時間情報の導入

- **タイムスタンプt**は**正規分布**によって生成　$t' \sim Gaussian(\mu_e, \sigma_e)$　$t \leftarrow \lfloor t' \rfloor$

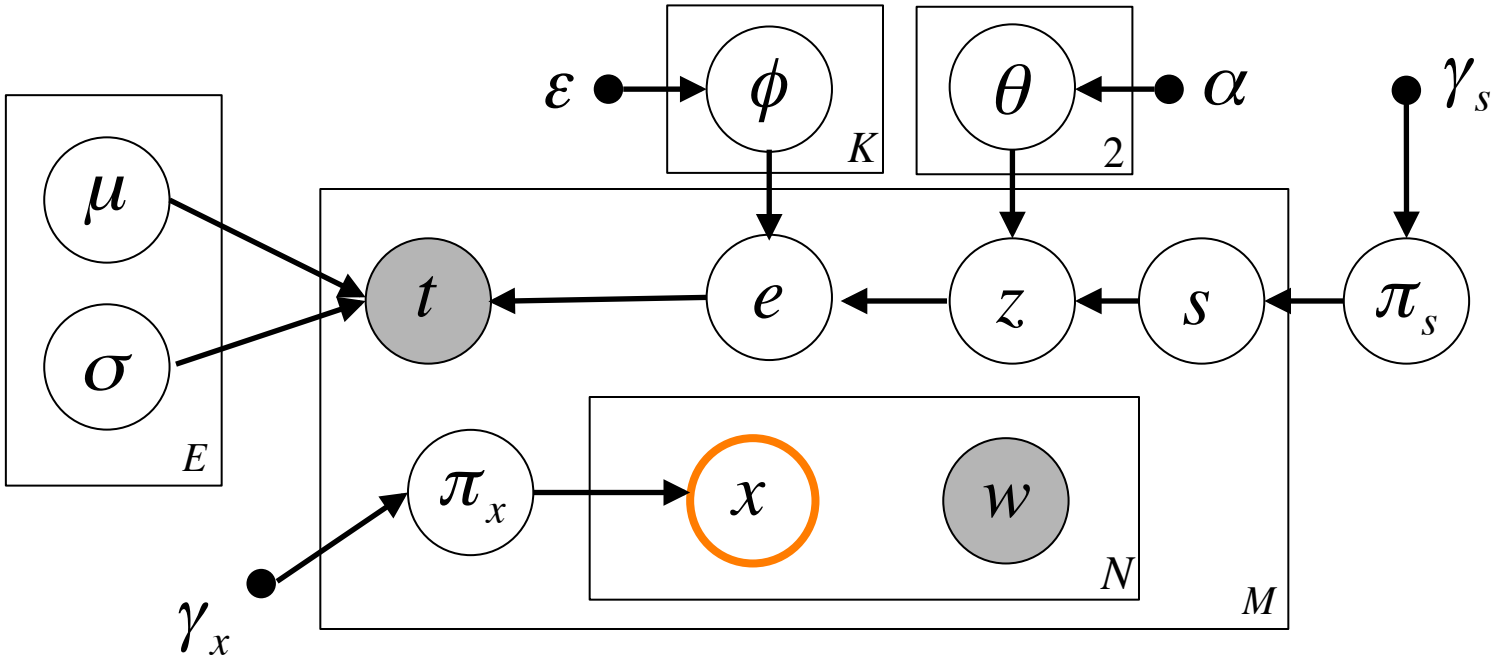




μから離れるほど文書はイベントeとの関連性が薄くなる

Time-aware Hierarchical Bayesian Model
トピック特有の情報とイベント特有の情報を区別

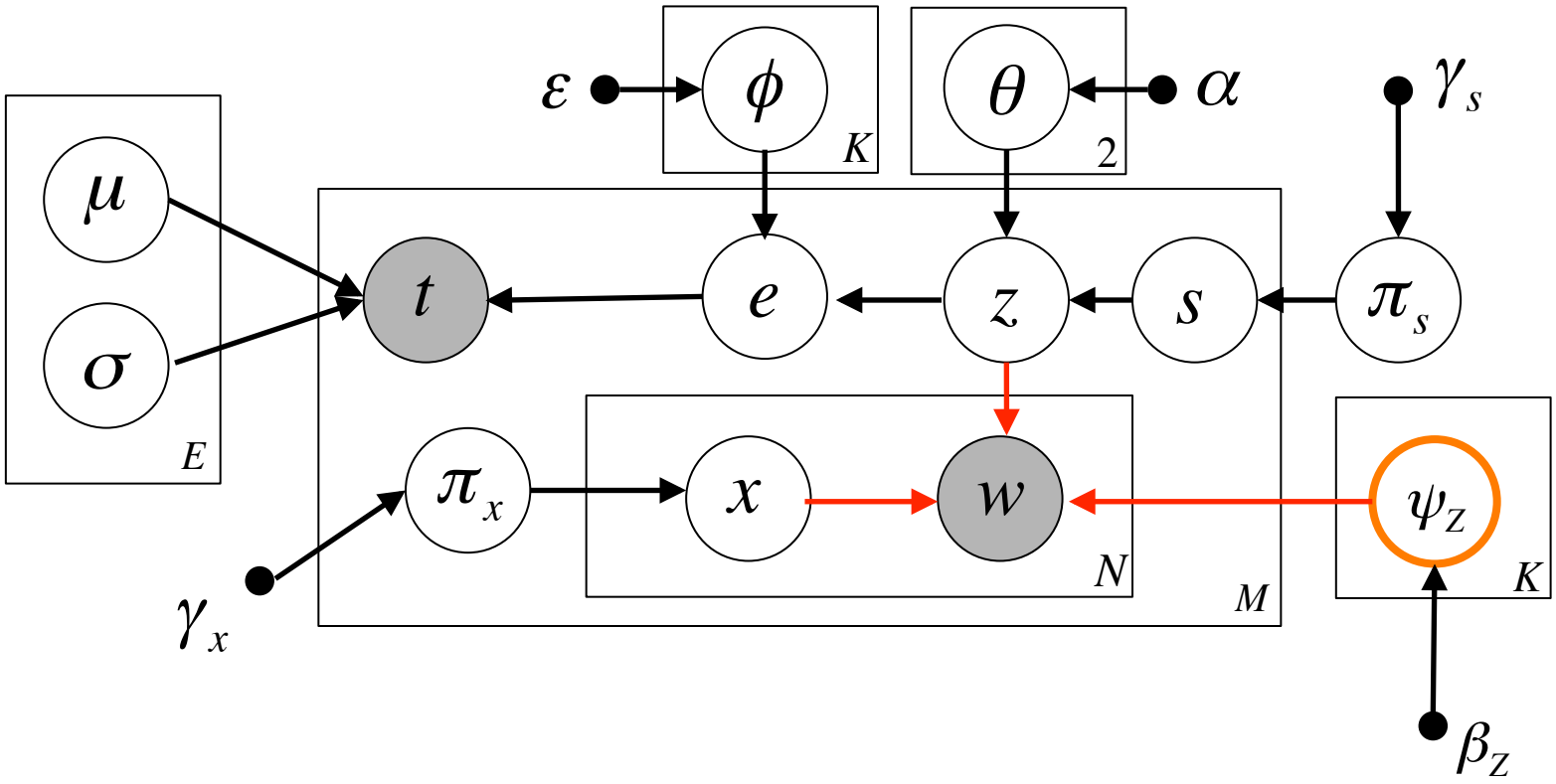
- 単語wがトピック特有単語か(x=0)イベント特有単語か(x=1)をベルヌーイ分布で決定する $x \sim Bernoulli(\pi_x)$ $\pi_x \sim Beta(\gamma_x)$



Time-aware Hierarchical Bayesian Model
トピック特有の情報とイベント特有の情報を区別

- x=0のときwはトピック特有単語分布より生成される

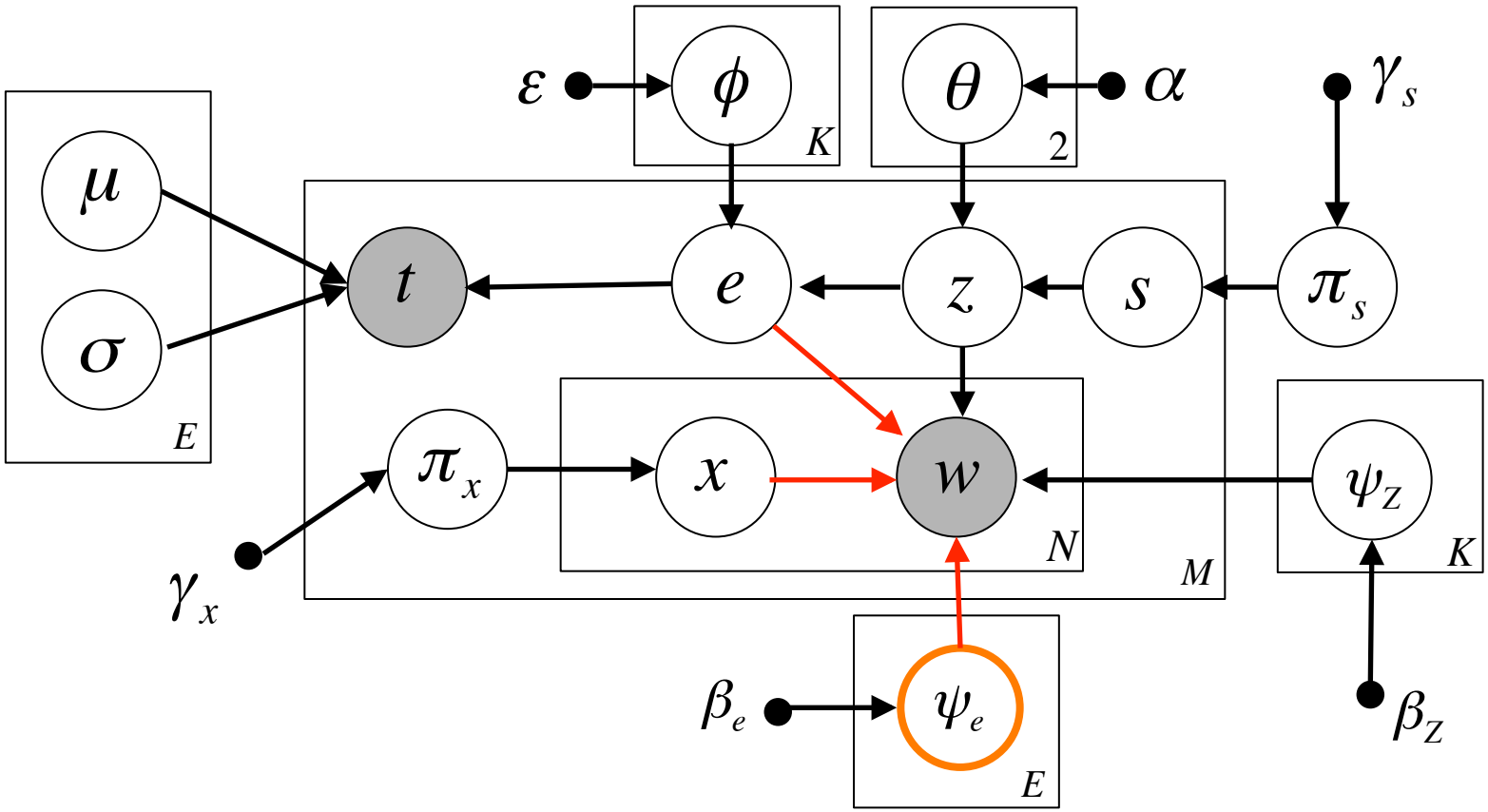
$$w \sim \psi_Z^{(z)} \quad \psi_Z^{(z)} \sim Dir(\beta_Z)$$



Time-aware Hierarchical Bayesian Model
トピック特有の情報とイベント特有の情報を区別

- x=1のときwはイベント特有単語分布より生成される

$$w \sim \psi_e^{(e)} \qquad \psi_e^{(e)} \sim Dir(\beta_e)$$



モデル推定

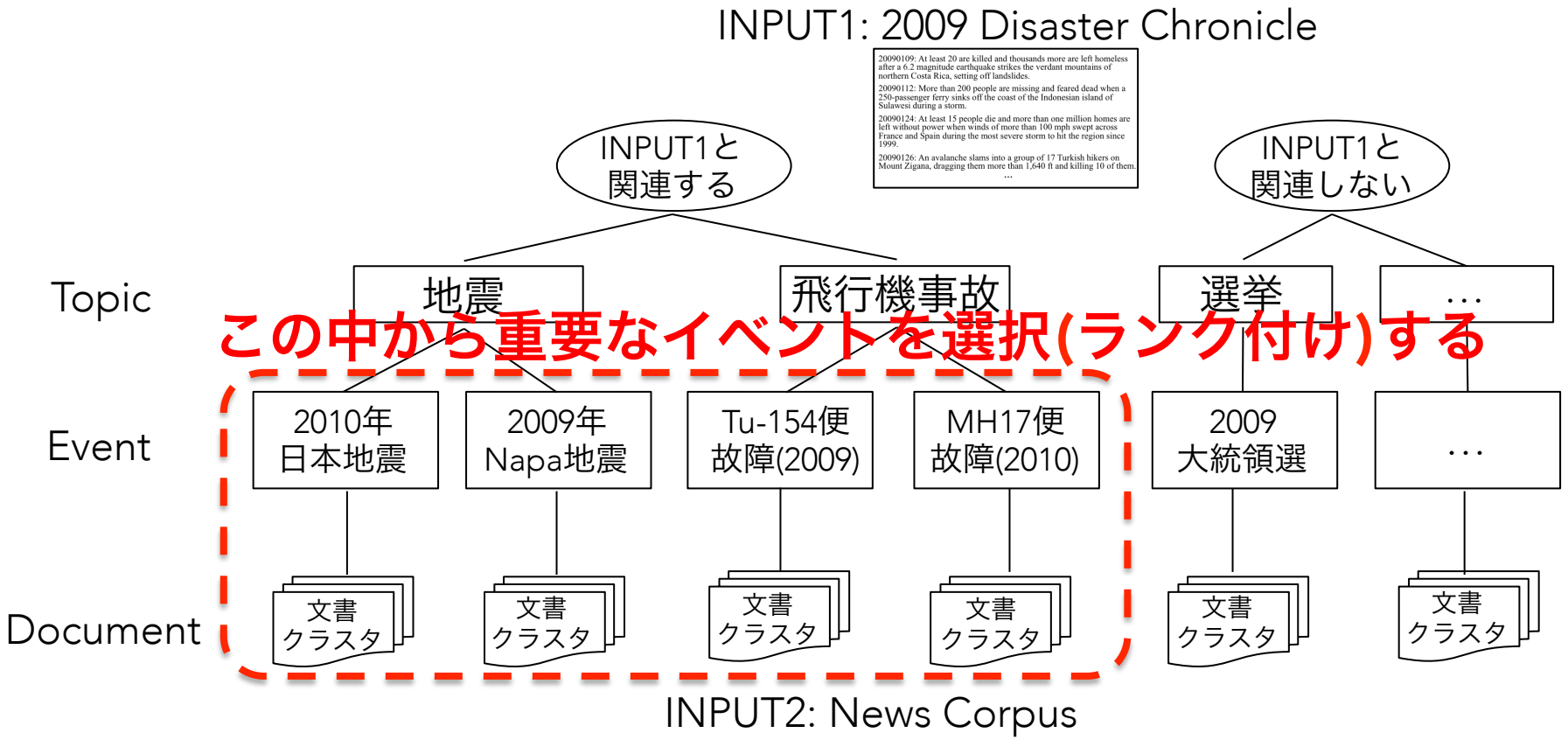
- **Collapsed ギブスサンプリング**
  - 潜在変数**s,z,e,x**を順番にサンプリング
    - サンプルのための確率式は省略
  - s,z,e,xに関するカウントを更新
  - タイムスタンプのガウス分布の平均μと分散σも更新

- **P(s)とP(z|s)はコーパスから推定できないので工夫する**
  - スポーツのChronicleと,災害のChronicleではP(s)，P(z|s)は異なる
  - コーパスの各文書にsを自動的にラベル付する
    - Lucineを使い，コーパス中の文書dと入力年代記内のイベントeの関連スコアsim(d,e)を計算し，高いものをs=1とする
    - sがラベル付けされた文書はその値をサンプルする
    - ラベル付けされていない文書は確率にもとづきサンプルする

STEP2：
SALIENT EVENT SELECTION

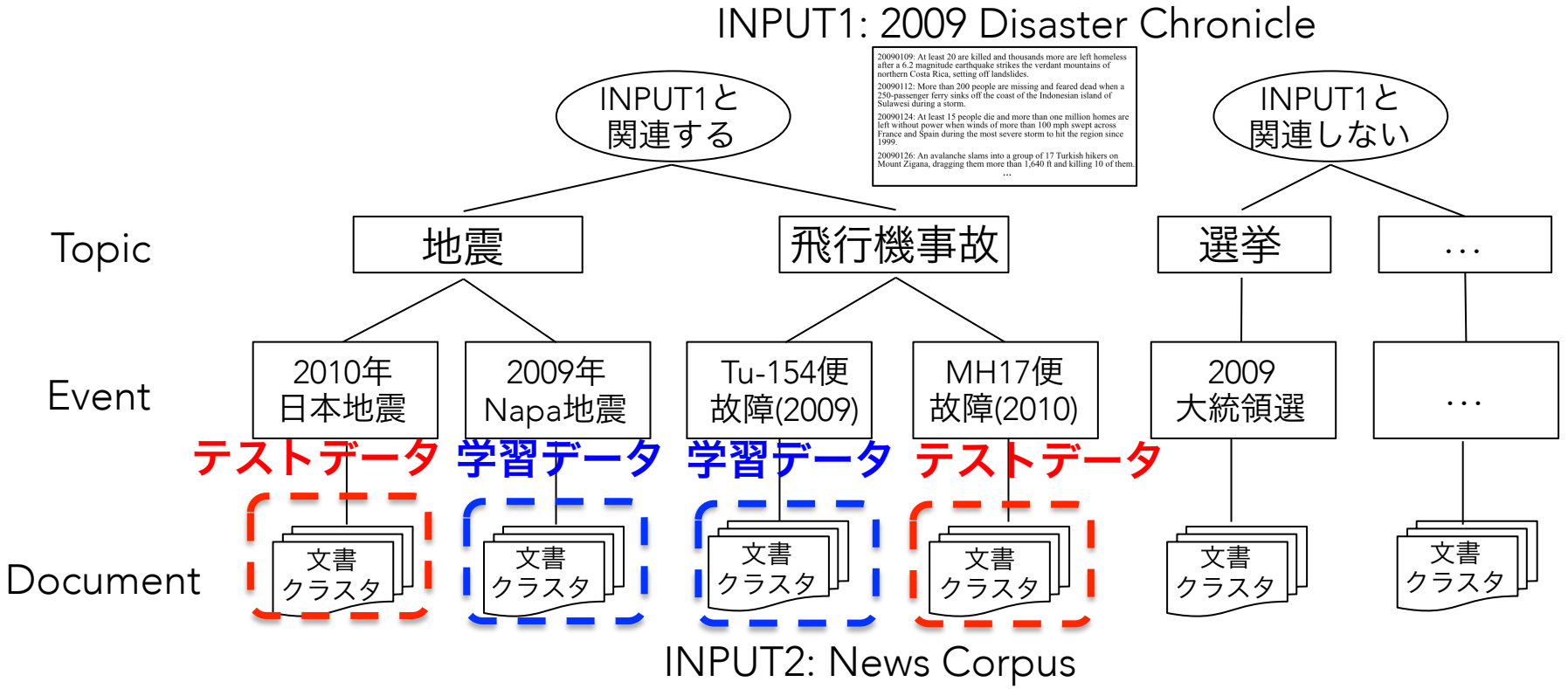
重要なイベントの選択(ランク付け)

EVENT DETECTIONにより下の構造が得られた



ランキング学習

- ランキング学習には**SVM-Rank**[Joachims, 2006]を使用する
- 学習データはEvent Detectionで得られた2009年の文書
- テストデータはEvent Detectionで得られた2010年の文書



SVM-Rankによるランキング学習

- **学習データの各イベントには優先度を与えておく必要がある**
  - 高優先度：入力Chronicleのイベントと関連が強いイベント
  - 低優先度：入力Chronicleのイベントと関連がないイベント
  - イベントをクエリとしたときの文書のLucineスコアで優先度を決定する

- **素性**
  - **P(s=1|e)**：入力年代記がイベントeと関連ある確率
  - **P(e|z)**：トピックzに対してイベントeのインパクトの確率
  - **σe**：eの正規分布の分散，つまりイベントeのスパン（どれだけの期間，影響があったか）
  - **|De|**：イベントeに関連すつ文書がいくつあったか，eのインパクト
  - **|De|/σe**：長期間イベントは|De|が多いのでその期間eで割ってノーマライズ

EXPERIMENTS

実験設定

- データ：2009年の年代記に対応する2010年の年代記を生成
  - 災害・スポーツ・戦争・政治と広範囲の2009年のChronicle
    - from mapreport, infoplease, wikipedia
  - English gigaword の2009-2010のAPWとXinhua newsを使う
    - タイトルと最初の段落にバーストワードを一つも持たない文書は除外
    - バーストワードはKleinbergのやつを使う
      - バースト状態とバーストでない状態を遷移させるモデル
    - 140557文書
- 前処理：Standard Core NLP でストップワードの削除
- パラメータ設定：
  - α=0.05 βz=0.005 βe=0.0001 γs=0.05 γx=0.5 ε=0.01 K=50 E=5000
  - ギブスサンプラーイテレーション2000回
  - 500回ごとにinference(推論)
  - SVM-rankの正規化パラメータc=0.1
- **イベントを表示するとき，どの文書を使うか (ヒューリスティック)**
  - **ニュース記事の最初の段落は良い要約になってる（傾向）**
  - **クラスタ内の一番最初に出た記事はイベントの説明になってる（傾向）**
    - **最初の記事のタイムスタンプをイベントの時間とする**

実験方法

- 評価用の正解Chronicleを作成
- 評価指標
  - Precision@k = $|E_g \cap E_{topk}| / k$
    - E_g: 正解Chronicleのイベント集合
    - E_{topk}: ランキングしたtop kのイベント集合
- ベースラインとなるランクスコア
    - BasicRankScore(e)：$\sum_d \max_e(\mathrm{sim}(d,e))$
- 比較手法
  - Random：ランダムにk文書選んでChronicleを生成
  - NB+basic：NBで推定したイベントに対してBasicRankScoreを計算
  - B-HAC+basic：一般ドメインでの最高性能のモデルburstVSM[Zhao et al., 2012]
  - TaHBM+basic：TaHBMで同定したイベント対してBasicRankScoreを計算
    - 提案した素性は使わない，ランキング学習の有用性評価のために追加

結果

- 提案手法が一番性能が良い

	sports		politics		disaster		war		comprehensive	
	P@50	P@100	P@50	P@100	P@50	P@100	P@50	P@100	P@50	P@100
Random	0.02	0.08	0	0	0.02	0.04	0	0	0.02	0.03
NB+basic	0.08	0.12	0.18	0.19	0.42	0.36	0.18	0.17	0.38	0.31
B-HAC+basic	0.10	0.13	0.30	0.26	0.50	0.47	0.30	0.22	0.36	0.32
TaHBM+basic	0.18	**0.15**	0.30	0.29	0.50	0.43	0.46	0.36	0.38	**0.33**
Our approach	**0.20**	**0.15**	**0.38**	**0.36**	**0.64**	**0.53**	**0.54**	**0.41**	**0.40**	**0.33**

- 生成結果
  - http://aclweb.org/anthology/attachments/P/P15/P15-1056.Notes.html

考察

- **災害Chronicle生成はよくて，スポーツChronicle生成はダメ**
  - スポーツは開幕戦や一回戦についての記事が多い(表示方法が原因)
  - 準々決勝以降の試合のみイベント記事として入力年代記にしるされていた?
  - スポーツ年代記は準決勝，決勝，優勝者の結果についての情報が必要
    - 初戦の結果などいらない
  - 災害の最速の文書は直接災害のイベントについて説明しているので高性能
  - それ以降はだいたい救助・追悼の言葉とかだったりする
  - 戦争も災害に近い傾向
  - 政治は複雑，政治のイベント（選挙とか）は予めアレンジされてる?（政府のシャットダウン）とかは予想できない
- **Comprehensive Chronicle生成はとても良いという訳でもない**
  - イベントがトピックに関連しているかどうか(s)を取り扱えなかったと考えられる
  - ランキング問題がBasic Rank Scoreで十分に解決できないぐらいの難易度?

エラー分析

- **4種類のエラー**
  - １：あるイベントのエントリのトピックがChronicleと無関係
  - ２：重要でないイベントが含まれている
    - 2010-08-28レバノンがカナダに勝った：開幕戦だから余り重要でない
  - ３：メジャーなイベントを直接説明していない
    - 2010-01-14トルコがハイチ地震の追悼の言葉を送った
  - ４：同じイベントが2つ存在する
- **原因の考察**
  - エラー１：detection時に関連するイベントかどうか判定するのに失敗している
  - もっと深刻なエラー：災害Chronicleはエラー3がある
    - 一つの災害に対して多くの記事（救助とか追悼とか）があるから
    - これらの返答はトピックとしては関連があり，多くの文書に含まれている
    - つまり，重要だと学習してトップリストとして出てきてしまう.
    - 解決策としては，モデルのパラメータを変えて，イベントの粒度を大きくする；メジャーなイベントを説明しているイベントと，そのイベントの返答を一つのクラスタとしてまとめ上げる
  - 一番深刻なエラー：スポーツChronicleはエラー2がある
    - 生成すべきChronicleでそのイベントが重要であるかどうかが替わる
    - スポーツだと書く試合についてだけど，compだとワールドカップは一つのイベント
    - つまり，年代記の違いによって，イベントの粒度を変える（適応）させる必要がある

イベントの発生時間の分析

- **イベントの発生時間はタイムスタンプの時間とは限らない**
  - 人手で編集した2010年のweb年代記の時間を正解とする
  - 各エントリの時間と, 実際のイベントの発生時間の差を評価
- **結果：**

sports	politics	disaster	war	comprehensive
0.800	3.363	1.042	1.610	2.467

  - スポーツ，災害，戦争の精度は良好
    - これらの重要イベントは即時にレポートされるため
  - 政治が一番悪い
    - 政治イベントの幾つかは機密事項で，発生から報道までラグがある
    - 他のいくつかのイベントはサミットとかはイベントが発生する前に報道される
  - Compは多く政治イベントが含まれるので精度が落ちる

まとめ

- **トピックが関連したChronicle生成**
  - Event Detection + Salient Event Selection
    - 階層的構造
    - 時間情報の活用
    - イベント特有の情報 と トピック特有の情報の区別
  - 言語に依存しない
  - ドメインに依存しない
  - 任意のトピックで拡張可能
  - 本タスクにおいて，最高性能
- **課題：**
  - イベントの粒度の自動適応

感想

- モデル自体はよくあるHBMに毛が生えた，という印象
- タイムスタンプを持つ文書なら応用可，他タスクでも使えそう
- この論文，多分面白いのは考察，エラー分析の方だった