Document Ranking in IR

- Given:
  - a set of documents $\mathcal{D} = \{d_1, \ldots, d_M\}$ and $|\mathcal{D}| = M$,
  - a set of terms $\mathcal{T} = \{t_1, \ldots, t_N\}$ and $|\mathcal{T}| = N$,
  - query terms $\vec{q}$ such that $\vec{q}$ is a subset in $\mathcal{T}$ (i.e., $\vec{q} \subset \mathcal{T}$).
- The score function $score(\vec{q}, d_i)$ returns a numeric score for any document d_i in the set $\mathcal{D}$ (that is, $d_i \in \mathcal{D}$) with respect to $\vec{q}$ and
- Top S elements from $\mathcal{D}$ with respect to the numeric scores will be selected (since the scores would enable us to completely order all the documents in $\mathcal{D}$).

Big Picture

- A **probabilistic graphical model (PGM)** is a probabilistic model for which a graph expresses the conditional dependence structure between random variables.
  - A **Bayesian belief network** is a PGM that represents a set of random variables and their conditional dependencies via a directed acyclic graph (DAG).
- A probabilistic **Information Retrieval** framework defines a specific ranking function, such as BM25, to rank documents according to their relevance to a given search query.
- Our goal: Combine Probabilistic Graphical Models, and Information Retrieval technologies to build up a Framework, for ranking documents.

Outline

- Basics of Bayesian Networks
- Basics of Document Ranking in Information Retrieval
- Our Model (BN+IR)
- Experimental Results
- Summary

Outline

- Basics of Bayesian Networks $\Leftarrow$
- Basics of Document Ranking in Information Retrieval
- Our Model (BN+IR)
- Experimental Results
- Summary

Example: Imagine a Family with a Dog...

We virtually always put the dog out (do) when the family is out (fo). We also put the dog out in the backyard for substantial periods of time when it has a (fortunately, infrequent) behavior problem (bp). A reasonable proportion of the time when the dog is out, you can hear her barking (hb) when you approach the house. We usually (but not always) leave the light on (lo) outside the house when the family is out. (Example due to Eugene Charniak)

Example: Imagine a Family with a Dog...

- Assign a node to each one of the five variables namely (FO, LO, BP, DO, HB) in the domain we have up to $2^5 = 32$ possible states of the world (note that for example, "fo" stands for "FO is true", whereas "$\neg$fo" stands for "FO is false"):

(fo, lo, bp, do, hb)
(fo, lo, bp, do, $\neg$hb)
(fo, lo, bp, $\neg$do, hb)
(fo, lo, bp, $\neg$do, $\neg$hb)
. . . . . .
($\neg$fo, $\neg$lo, $\neg$bp, $\neg$do, hb)
($\neg$fo, $\neg$lo, $\neg$bp, $\neg$do, $\neg$hb)

Example: Imagine a Family with a Dog...

- We do not know the true state of the world and one can only assign a degree of belief (between 0 to 1) to each one of these 32 states:

$$
\begin{aligned}
&Pr(fo, lo, bp, do, hb) = ?\\
&Pr(fo, lo, bp, do, \neg hb) = ?\\
&Pr(fo, lo, bp, \neg do, hb) = ?\\
&Pr(fo, lo, bp, \neg do, \neg hb) = ?\\
&\ldots\ldots\ldots\\
&Pr(\neg fo, \neg lo, \neg bp, \neg do, hb) = ?\\
&Pr(\neg fo, \neg lo, \neg bp, \neg do, \neg hb) = ?
\end{aligned}
$$

Example: Imagine a Family with a Dog...

- It is obvious that the sum over beliefs on all states is 1.

$$\Pr(\text{fo}, \text{lo}, \text{bp}, \text{do}, \text{hb}) = ?$$
$$\Pr(\text{fo}, \text{lo}, \text{bp}, \text{do}, \neg\text{hb}) = ?$$
$$\Pr(\text{fo}, \text{lo}, \text{bp}, \neg\text{do}, \text{hb}) = ?$$
$$\Pr(\text{fo}, \text{lo}, \text{bp}, \neg\text{do}, \neg\text{hb}) = ?$$
$$\ldots\ldots\ldots$$
$$\Pr(\neg\text{fo}, \neg\text{lo}, \neg\text{bp}, \neg\text{do}, \text{hb}) = ?$$
$$\Pr(\neg\text{fo}, \neg\text{lo}, \neg\text{bp}, \neg\text{do}, \neg\text{hb}) = ?$$

$$\text{sum} = 1.0$$

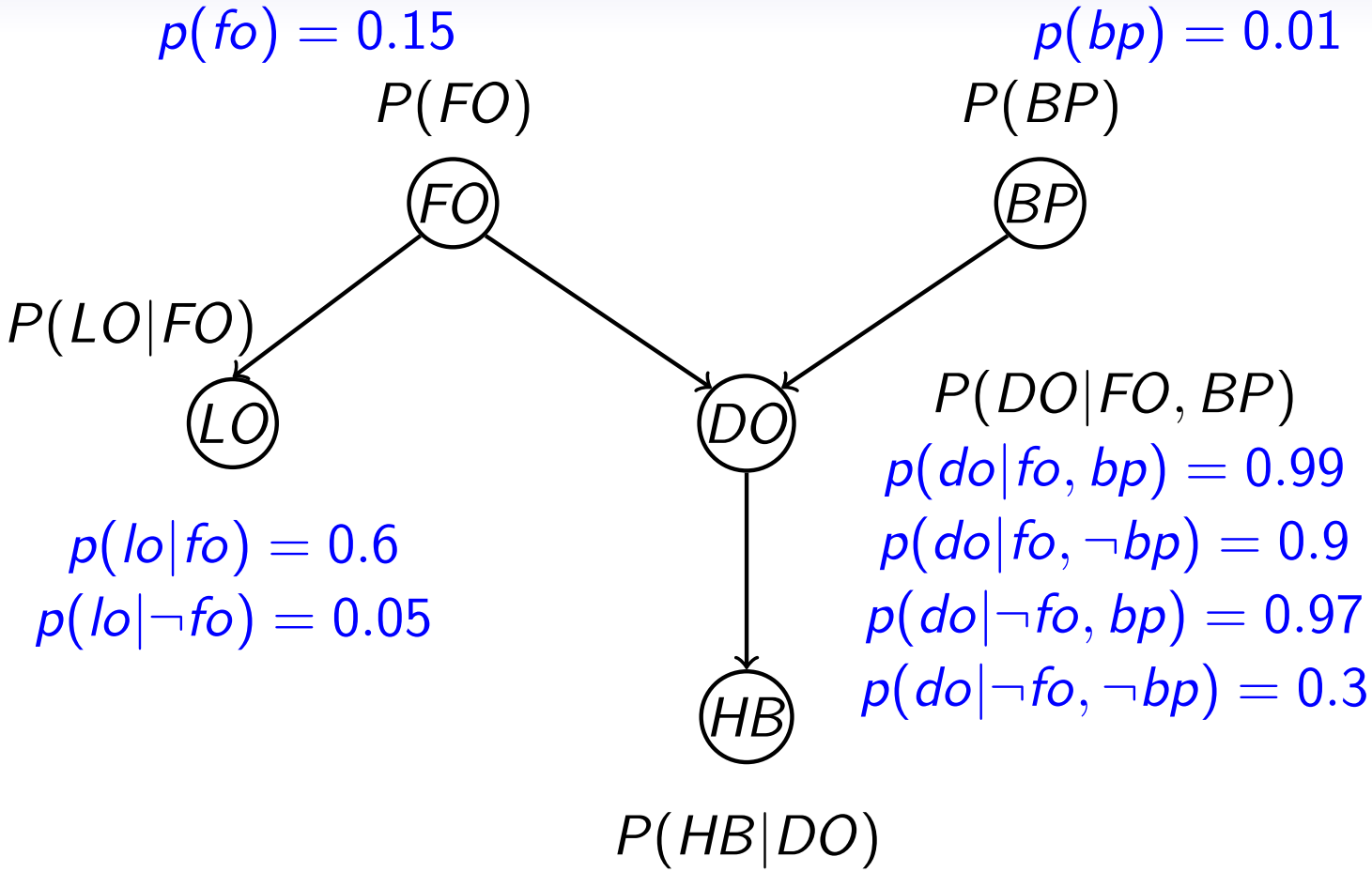
The Challenge

- We need to specify $2^5 = 32$ numbers. But when n is getting larger, this is not workable.
- Alternatively we can assume that all variables are conditionally independent to each other, thus we only need to specify 5 numbers, and any state can be simply calculated as a product of five values.
- For example, assume that
  Pr(fo)=0.15, Pr(lo)=0.6, Pr(bp)=0.1, Pr(do)=0.4,
  Pr(hb)=0.5,
  Pr(fo, lo, bp, $\neg$do, $\neg$hb)=
  $0.15 \times 0.6 \times 0.1 \times (1 - 0.4) \times (1 - 0.5) = 0.0027$.
- But this assumption is too extreme - typically there will be dependency among some variables.

Better Idea: Bayesian Networks

- A Bayesian Network is a directed acyclic graph;
- Nodes in BNs correspond to random variables;
- Pairs of nodes in a BN graph might be connected by a directed edge;
- Effect of the parents of a node X is quantified by a conditional probability distribution $P(X|Parents(X))$;
- That is to say that, each variable is conditionally independent from the non-parent variables given the parent variables.

Family with a Dog in Bayesian Networks



Bayesian Networks: How It Works

- It follows that

$$Pr(FO, LO, BP, DO, HB) =$$

$$Pr(FO) \times Pr(LO|FO) \times Pr(BP) \times Pr(DO|FO, BP) \times Pr(HB|DO);$$

- For example, $Pr(fo, lo, bp, \neg do, \neg hb) =$

$$Pr(fo) \times Pr(lo|fo) \times Pr(bp) \times (1 - Pr(do|fo, bp)) \times (1 - Pr(hb|\neg do)) =$$

$$0.15 \times 0.6 \times 0.01 \times 0.01 \times 0.99 = 0.00000891$$

- BNs are often used to carry out probabilistic inference: computing posterior distribution of a set of variables given a set of *evidence variables* – variables whose values are observed or assigned;
- Suppose we want to use the BN to calculate the probability that the family is out given that the light is on but we did not hear barking: $Pr(fo|lo, \neg hb)$. See next page ...

Exact Inference on Posterior $Pr(fo|lo, \neg hb)$

- Using the conditional probabilities we have

$$Pr(fo|lo, \neg hb) = \frac{Pr(fo, lo, \neg hb)}{Pr(lo, \neg hb)}$$

- To compute, we need the following eight elements,
  $Pr(fo, lo, bp, do, \neg hb) = .15 \times .6 \times .01 \times .99 \times .3 = .0002673$
  $Pr(fo, lo, bp, \neg do, \neg hb) = .15 \times .6 \times .01 \times .01 \times .99 = .00000891$
  $Pr(fo, lo, \neg bp, do, \neg hb) = .15 \times .6 \times .99 \times .9 \times .3 = .024057$
  $Pr(fo, lo, \neg bp, \neg do, \neg hb) = .15 \times .6 \times .99 \times .1 \times .99 = .0088209$
  $Pr(\neg fo, lo, bp, do, \neg hb) = .85 \times .05 \times .01 \times .97 \times .3 = .000123675$
  $Pr(\neg fo, lo, bp, \neg do, \neg hb) = .85 \times .05 \times .01 \times .03 \times .99 = .0000126225$
  $Pr(\neg fo, lo, \neg bp, do, \neg hb) = .85 \times .05 \times .99 \times .3 \times .3 = .00378675$
  $Pr(\neg fo, lo, \neg bp, \neg do, \neg hb) = .85 \times .05 \times .99 \times .7 \times .99 = .029157975$
- $Pr(fo, lo, \neg hb)$ is the sum of the first four values (0.003315411), and $Pr(lo, \neg hb)$ is the sum of all values (0.00662369056); hence,

$$Pr(fo|lo, \neg hb) = \frac{Pr(fo, lo, \neg hb)}{Pr(lo, \neg hb)} = \frac{0.003315411}{0.00662369056} \approx 0.5005$$

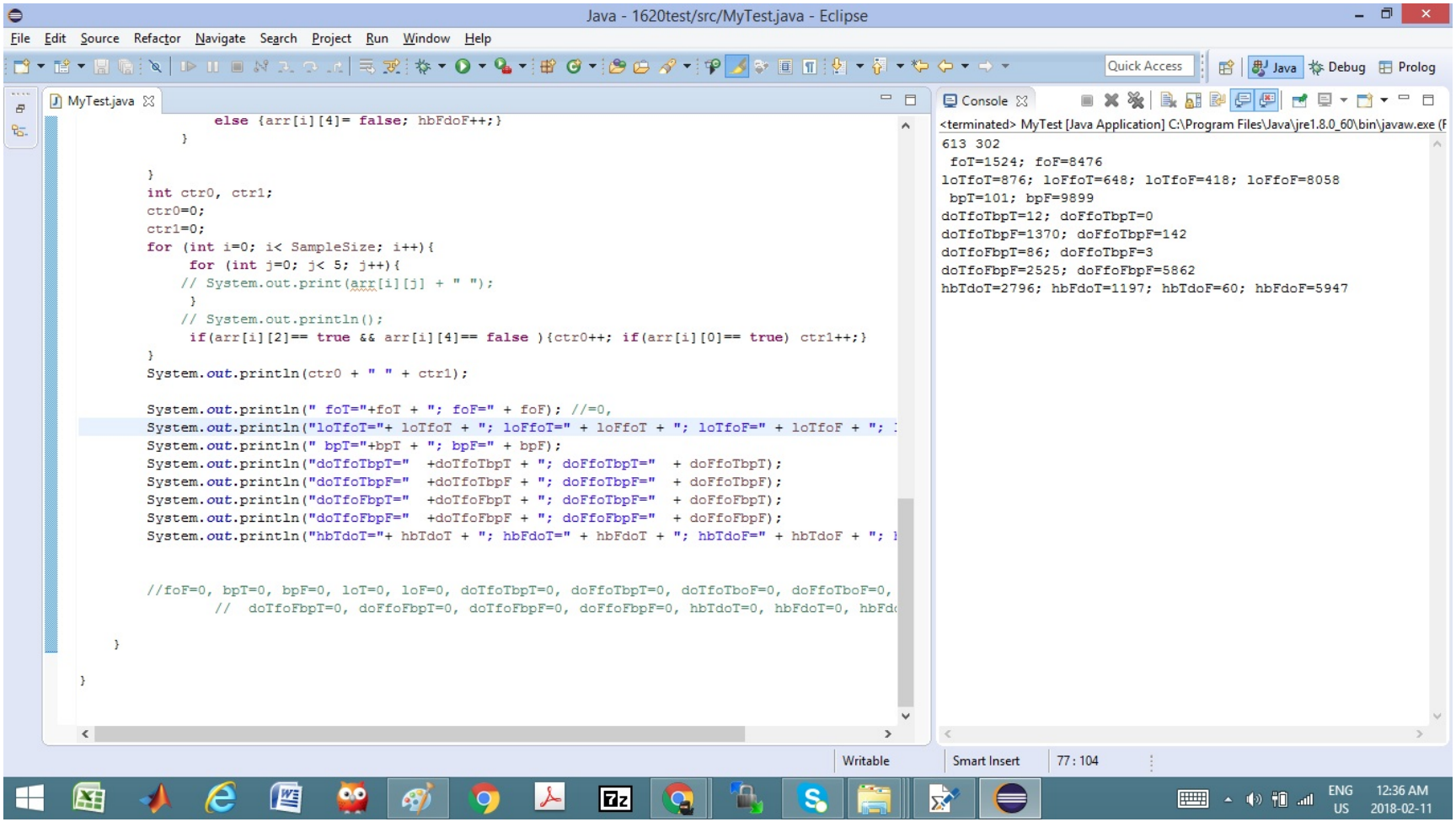
Approximate Inference on Posterior: Rejection Sampling

- Exact inference in large networks is computationally intractable, thus it is essential to consider approximate inference methods (for example, rejection sampling);
- Samples are generated from known probability distributions;
- Given a random number generator uniformly distributed in the range [0,1], we can sample on a single variable;
- Sampling on BNs based on the idea to sample variables in a given net on topological order;
- The probability distribution from which the value is sampled is conditioned on the values already assigned to the variable's parents.

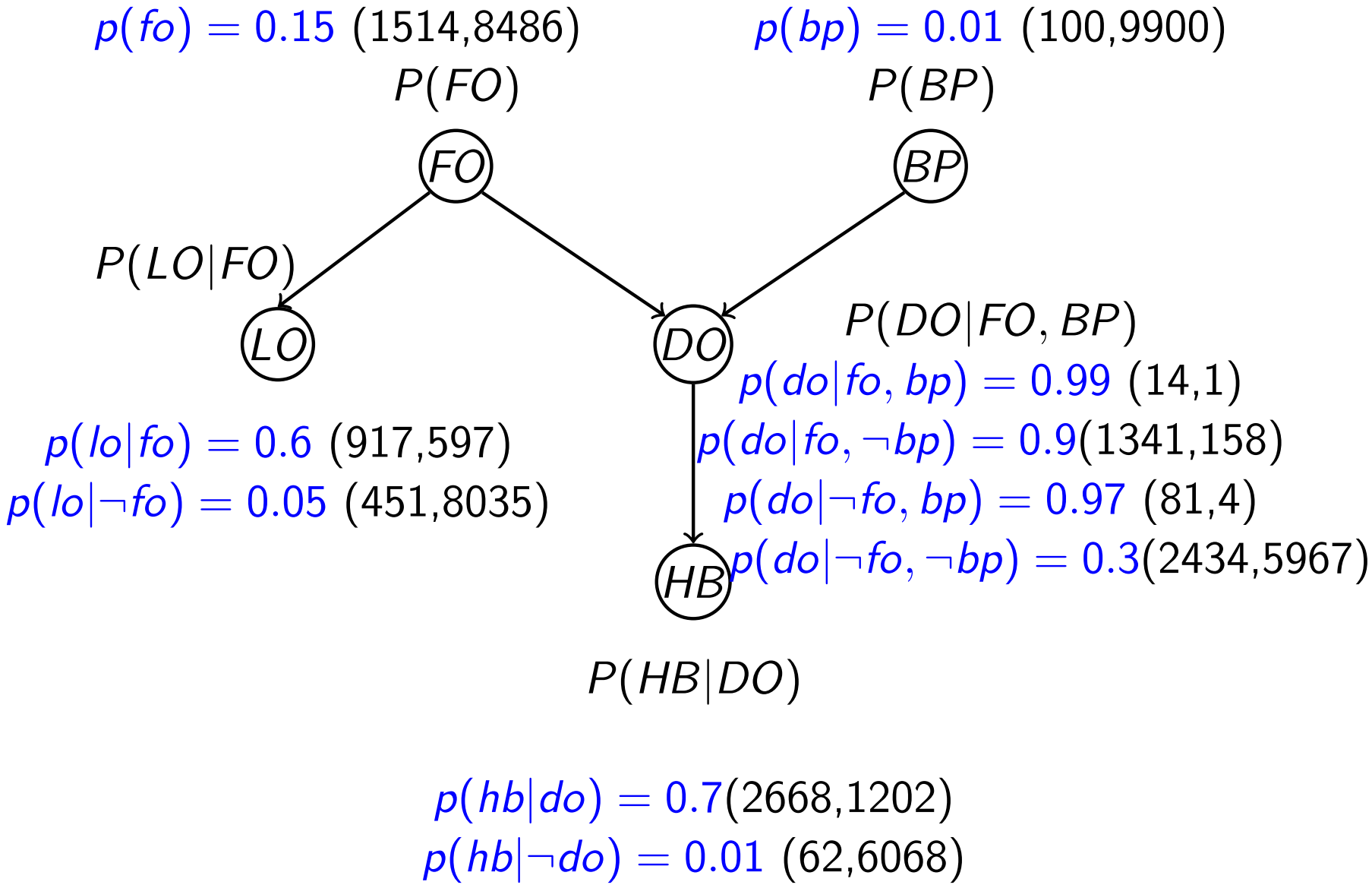
Rejection Sampling on the BN of Family/Dog

- Assuming an ordering of (FO, LO, BP, DO, HB):
  1. Sample from $P(FO) = \langle 0.15, 0.85 \rangle$, value is $\neg fo$,
  2. Sample from $P(LO|\neg fo) = \langle 0.05, 0.95 \rangle$, value is $\neg lo$,
  3. Sample from $P(BP) = \langle 0.01, 0.99 \rangle$, value is $\neg bp$,
  4. Sample from $P(DO|\neg fo, \neg bp) = \langle 0.3, 0.7 \rangle$, value is do,
  5. Sample from $P(HB|do) = \langle 0.7, 0.3 \rangle$, value is $\neg hb$,
- In this case the sample creates an event of $(\neg fo, \neg lo, \neg bp, do, \neg hb)$.

Let Us Run 10,000 Samples



Let Us Run 10,000 Samples



Rejection Sampling for Estimating $Pr(fo|lo, \neg hb)$

- $Pr(fo|lo, \neg hb)$ can actually be estimated as the ratio of

$$\frac{\#\text{ of samples fo, lo, and not hb}}{\#\text{ of samples lo, and not hb}} = \frac{336}{668} = 0.50299,$$

  which is pretty close to true value 0.5005.
- When number of sample increases, the approximation is getting closer to the exact value.
- For example, when we run 1 million samples, the ratio is 32996/65858, which is 0.5010, even closer to 0.5005.

Outline

- Basics of Bayesian Networks
- Basics of Document Ranking in Information Retrieval $\Leftarrow$
- Our Model (BN+IR)
- Experimental Results
- Summary

Document Ranking in IR

- Given:
  - a set of documents $\mathcal{D} = \{d_1, \ldots, d_M\}$ and $|\mathcal{D}| = M$,
  - a set of terms $\mathcal{T} = \{t_1, \ldots, t_N\}$ and $|\mathcal{T}| = N$,
  - query terms $\vec{q}$ such that $\vec{q}$ is a subset in $\mathcal{T}$ (i.e., $\vec{q} \subset \mathcal{T}$).
- The score function $score(\vec{q}, d_i)$ returns a numeric score for any document d_i in the set $\mathcal{D}$, and
- Top S elements from $\mathcal{D}$ with respect to the numeric scores will be selected;
- Baseline $score(\vec{q}, d)$ can be defined as $\sum_{t \in \vec{q}} (tf_{t,d} \times idf_t)$, where
  - $tf_{t,d}$ denote *term frequency*, the number of occurrences of term t in document d,
  - df_t denotes *document frequency*, the number of documents in $\mathcal{D}$ that contain the term t, and
  - $idf_t = \log(M/df_t)$ denotes *inverse document frequency*.

BM25

Specifically, BM25 score function $score_{bm25}(\vec{q}, d)$ is defined as

$$\sum_{t \in \vec{q}} idf_t^{bm25} \cdot \frac{(k_1+1)tf_{t,d}}{k_1\left((1-b)+b\cdot\left(\frac{|d|}{avdl}\right)\right)+tf_{t,d}} \cdot \frac{(k_3+1)tf_{t,\vec{q}}}{k_3+tf_{t,\vec{q}}}, \text{where}$$

- $|d|$ denotes the length of the document d, and $avdl$ denotes the average length of documents in $\mathcal{D}$;
- constant $k_1 > 0$ controls the scaling of $tf_{t,d}$;
- constant $k_3 > 0$ controls the scaling of $tf_{t,\vec{q}}$;
- constant $b \in [0, 1]$ controls the level of document normalization; When $b = 0$, normalization is not in effect, and when $b = 1$, full level of normalization proportional to the ratio of $\frac{|d|}{avdl}$ is applied, and
- $idf_t^{bm25} = \log \frac{M - df_t + 0.5}{df_t + 0.5}$ is a variant to idf_t.

Outline

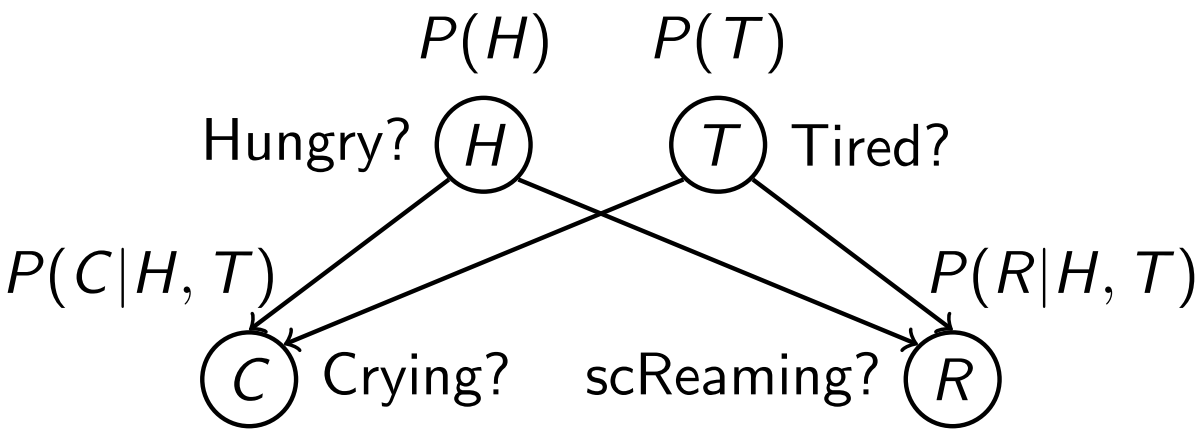
- Basics of Bayesian Networks
- Basics of Document Ranking in Information Retrieval
- Our Model (BN+IR) $\Leftarrow$
- Experimental Results
- Summary

Bayesian Networks: Crying Baby

Consider a simple example of *Baby World*, which consists of four random variables H, T, C, and R, corresponding to the variable facts that the baby is *Hungary*, *Tired*, *Crying*, and *scReaming*, respectively.

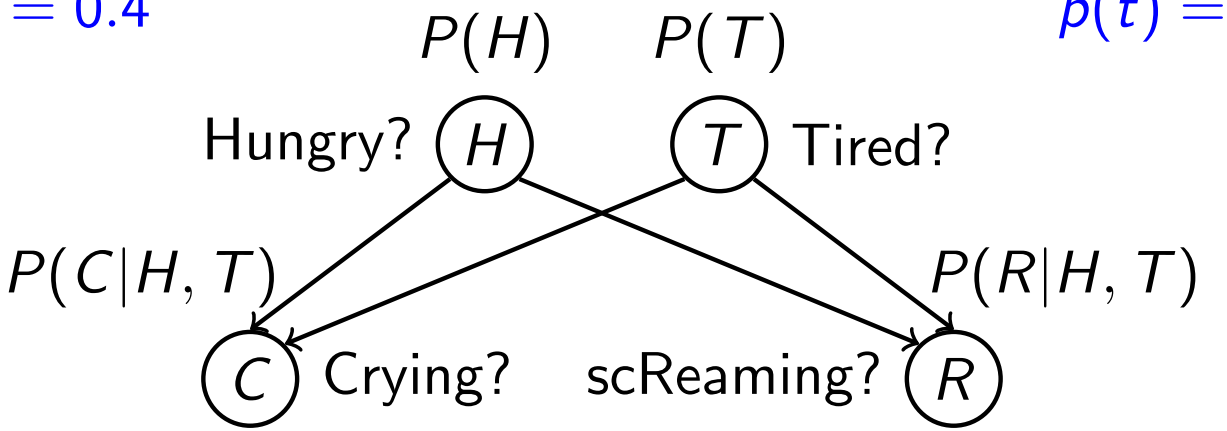
After all four conditional probability distributions are specified, we could compute, for example, the probability of *Tired* if we observe it *Crying* but do not hear it *scReaming*:

$$P(T \text{ is true} \mid C \text{ is true, and } R \text{ is false})$$



Conditional Probability Distributions

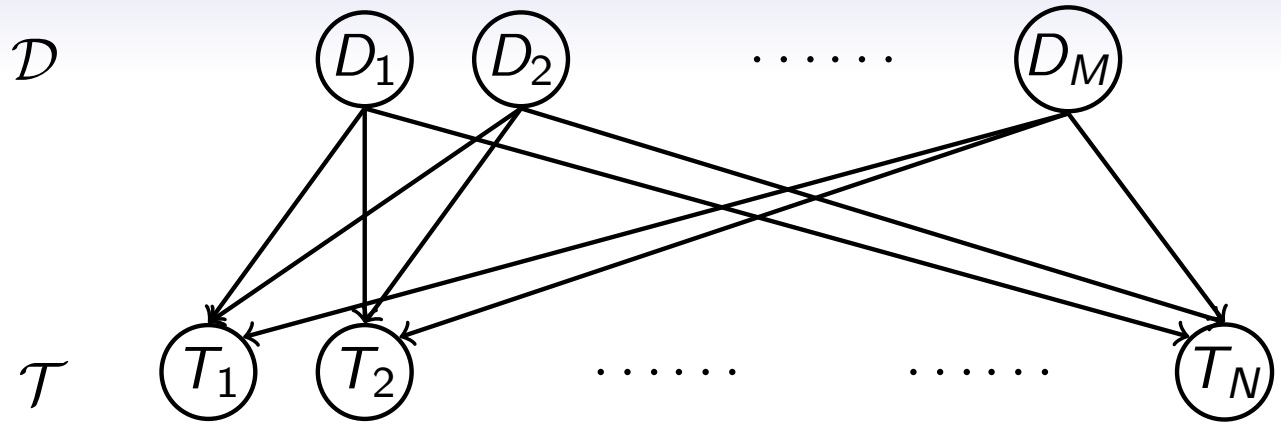
$p(h) = 0.4$ $p(t) = 0.2$



$$p(c|h,t) = 0.95 \quad p(r|h,t) = 0.95$$
$$p(c|h,\neg t) = 0.9 \quad p(r|h,\neg t) = 0.1$$
$$p(c|\neg h,t) = 0.9 \quad p(r|\neg h,t) = 0.9$$
$$p(c|\neg h,\neg t) = 0.05 \quad p(r|\neg h,\neg t) = 0.05$$

$P(T \text{ is true} \mid C \text{ is true, and } R \text{ is false}) = 0.05$

A BN Model for Document Ranking: Topology



- The BN is a two-layer directed graph containing a node in the top layer for each document variable D_i and a node in the bottom layer for each term variable T_j (D_i selected with respect to a query, and T_j is a query term),
- Each $D_i \in \mathcal{D}$ takes two values: d_i^1, d_i^0, representing "D_i selected with respect to a query" (d_i^1) or not (d_i^0),
- Each $T_j \in \mathcal{T}$ takes two values: t_j^1, t_j^0, representing "T_j is a query term" (t_j^1) or not (t_j^0),
- An edge from D_i to T_j represents that term T_j appears in the document D_i. It is assumed no edges between document variables in $\mathcal{D}$, and no edges between term variables in $\mathcal{T}$.

A BN Model for Document Ranking: CPD

- $P(D_i)$, the prior distribution over a document D_i, equals the ratio of the size of the set of selected documents $\mathcal{S}$ to the size of $\mathcal{D}$, i.e., $P(D_i) = S/M$ for all documents,
- D_i, and $P(T_j|D1, \ldots, D_M)$, the conditional probability distribution of T_j given D_1, D_2, ..., and D_M
  - totally 2^M of them,
  - most probabilistic inference problems in BNs are NP-hard,
  - solution: estimating posteriors through statistical sampling (rejection sampling).

Statistical Sampling

- Suppose we have a set of samples $\mathcal{P}$ where $|\mathcal{P}| = P$, for any sample P^j the expected number of documents in P^j is S and these documents are put in the set $\mathcal{S}^j$, hence $|\mathcal{S}^j| = S$;
- P^j is accepted iff $score_{bn}(\vec{q}, \mathcal{S}^j) \geq thld$, and if P^j is accepted, it is added to the set $\mathcal{P}^{accepted}$;
  - Assuming BM25 $score(\vec{q}, d)$ is known for all $d \in \mathcal{D}$, given again P^j, we can straightforwardly instantiate the above inequality by respectively taking averages on scores over the set $\mathcal{S}^j$ and the set $\mathcal{D}$, and then comparing the two averages:

$$\frac{\sum_{d \in \mathcal{S}^j} score_{bm25}(\vec{q}, d)}{S} \geq \frac{\sum_{d \in \mathcal{D}} score_{bm25}(\vec{q}, d)}{M}$$

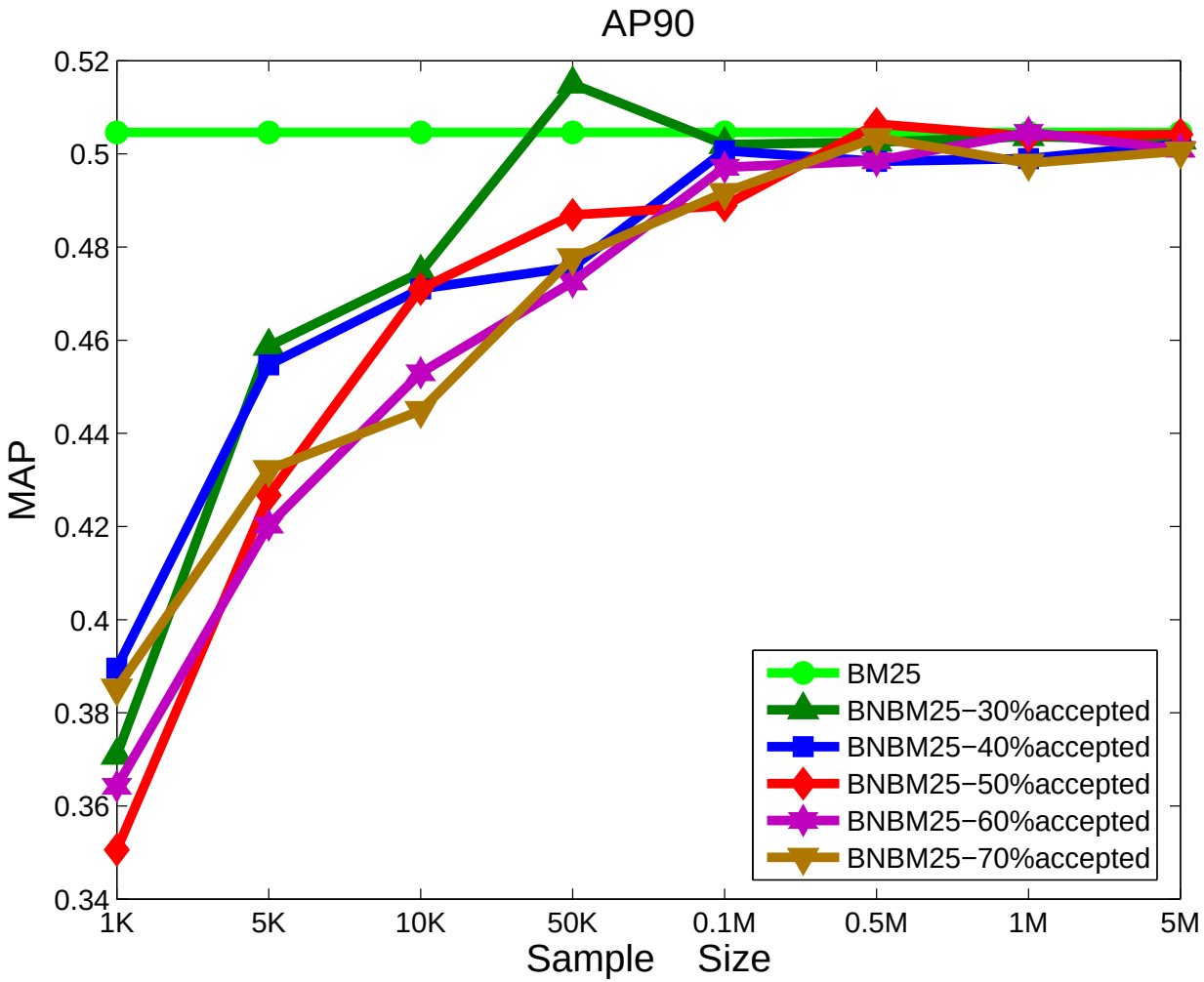
- Eventually when sampling finishes, the number of samples in $\mathcal{P}^{accepted}$ containing d can be counted out. This number is used to rank d.

Experimental Results: Test Collections Used

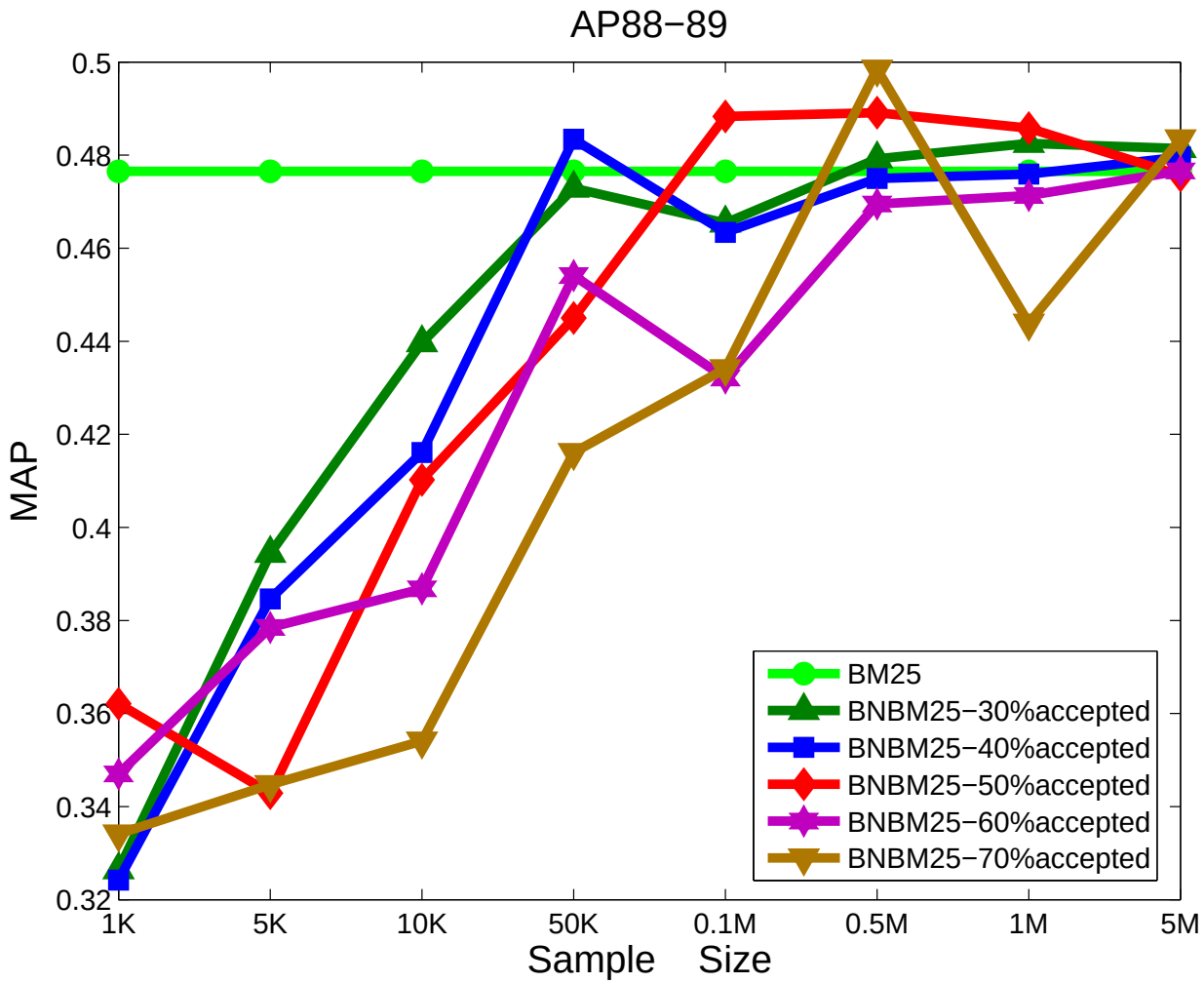
We conduct experiments on eight standard test collections including AP90, AP88-89, AP88-90, SJMN, and etc. Sizes and genres of them are different in general:

Collection	#of Docs	Size	Single Term Queries
AP90	78,321	0.23Gb	57, 75, 77, 78
AP88-89	164,597	0.50Gb	57, 75, 77, 78
AP88-90	242,918	0.73Gb	57, 75, 77, 78
SJMN	90,257	0.29Gb	57, 75, 77, 78
FT	210,158	0.56Gb	349, 364, 367, 392, 395
LA	131,896	0.48Gb	349, 364, 367,392, 395
TREC8	528,155	1.85Gb	403, 417, 424
WT2G	247,491	2.14Gb	403, 417, 424

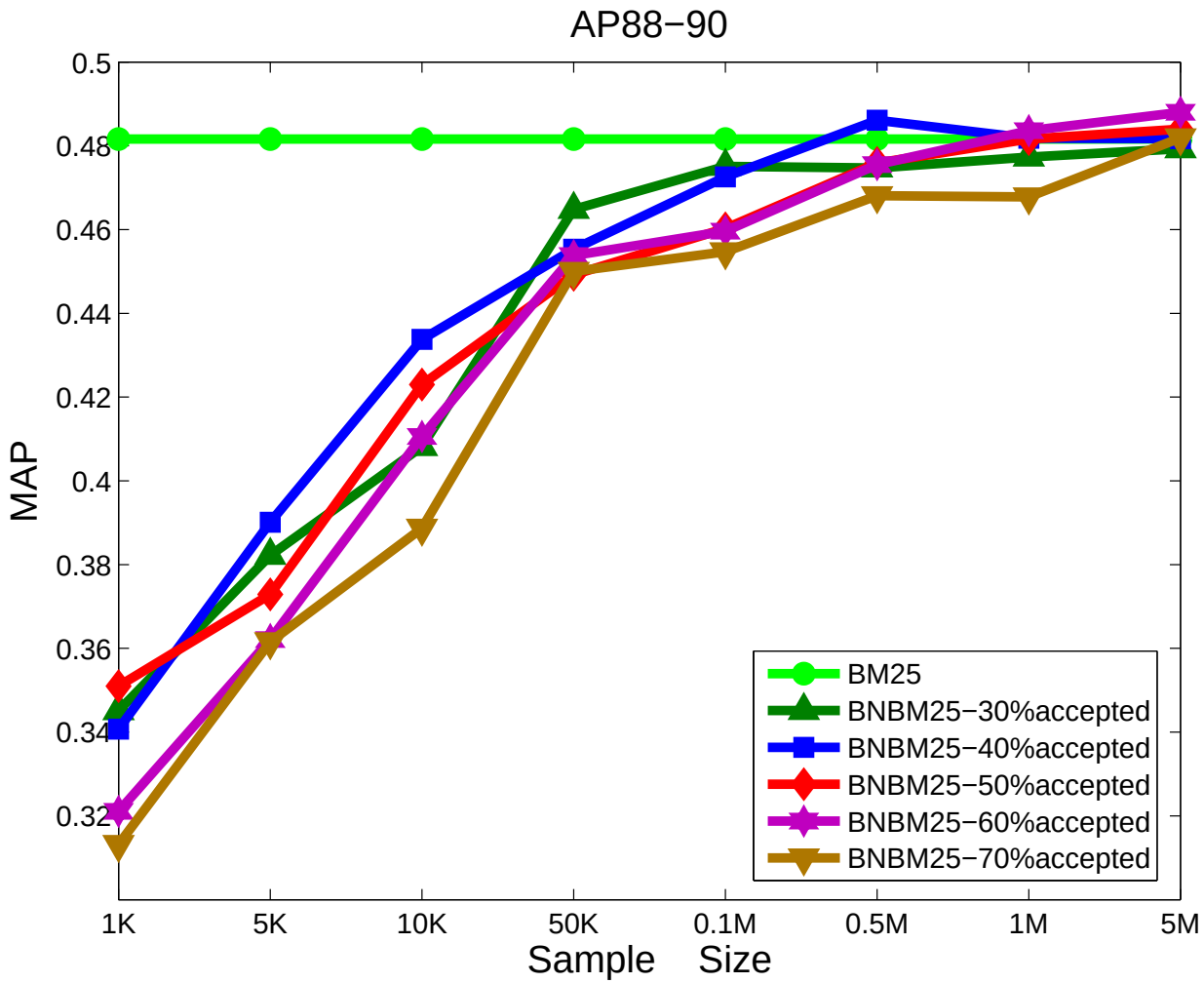
Experimental Results: Comparisons of BNBM25 and BM25



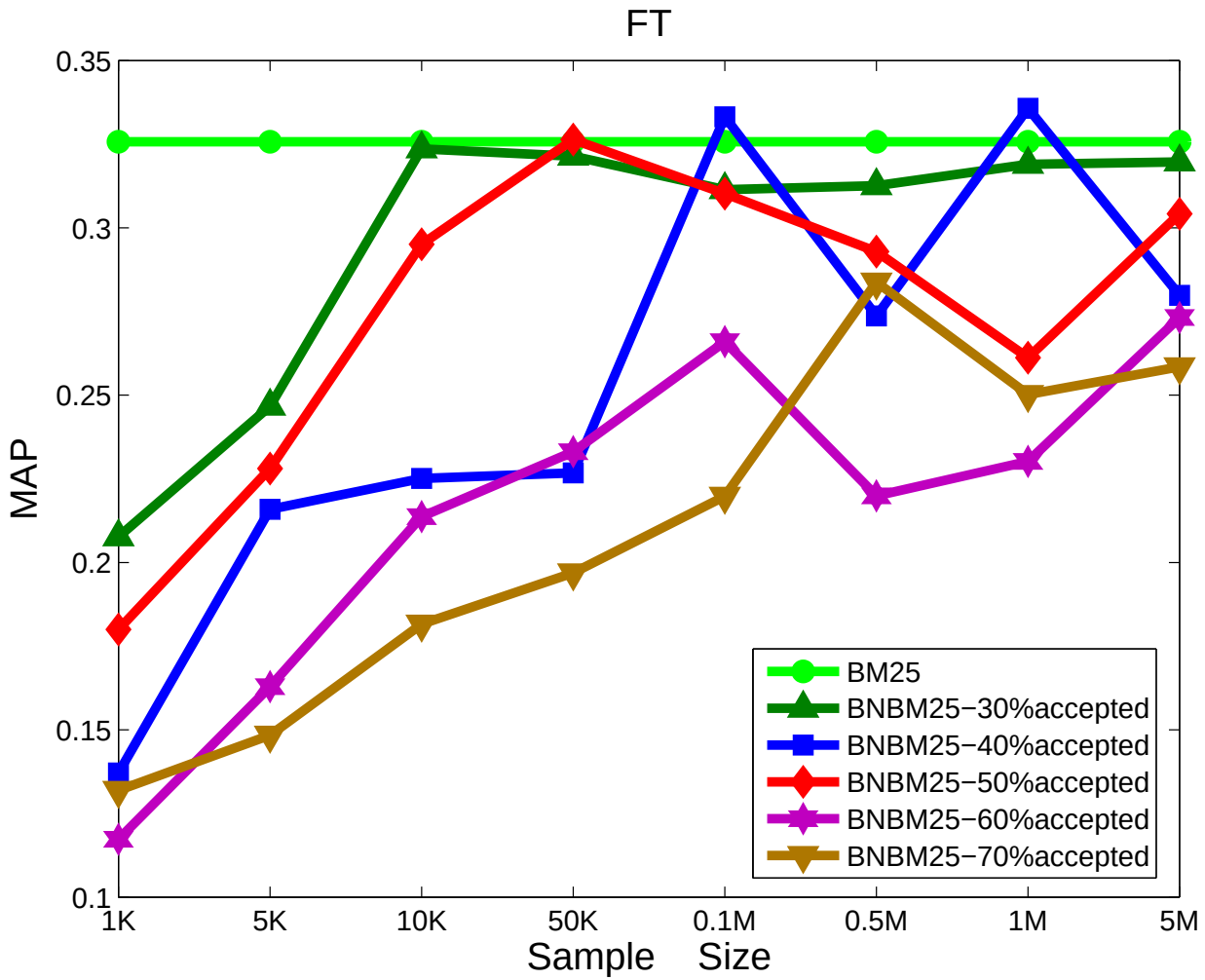
Experimental Results: Comparisons of BNBM25 and BM25 (Cont'd)



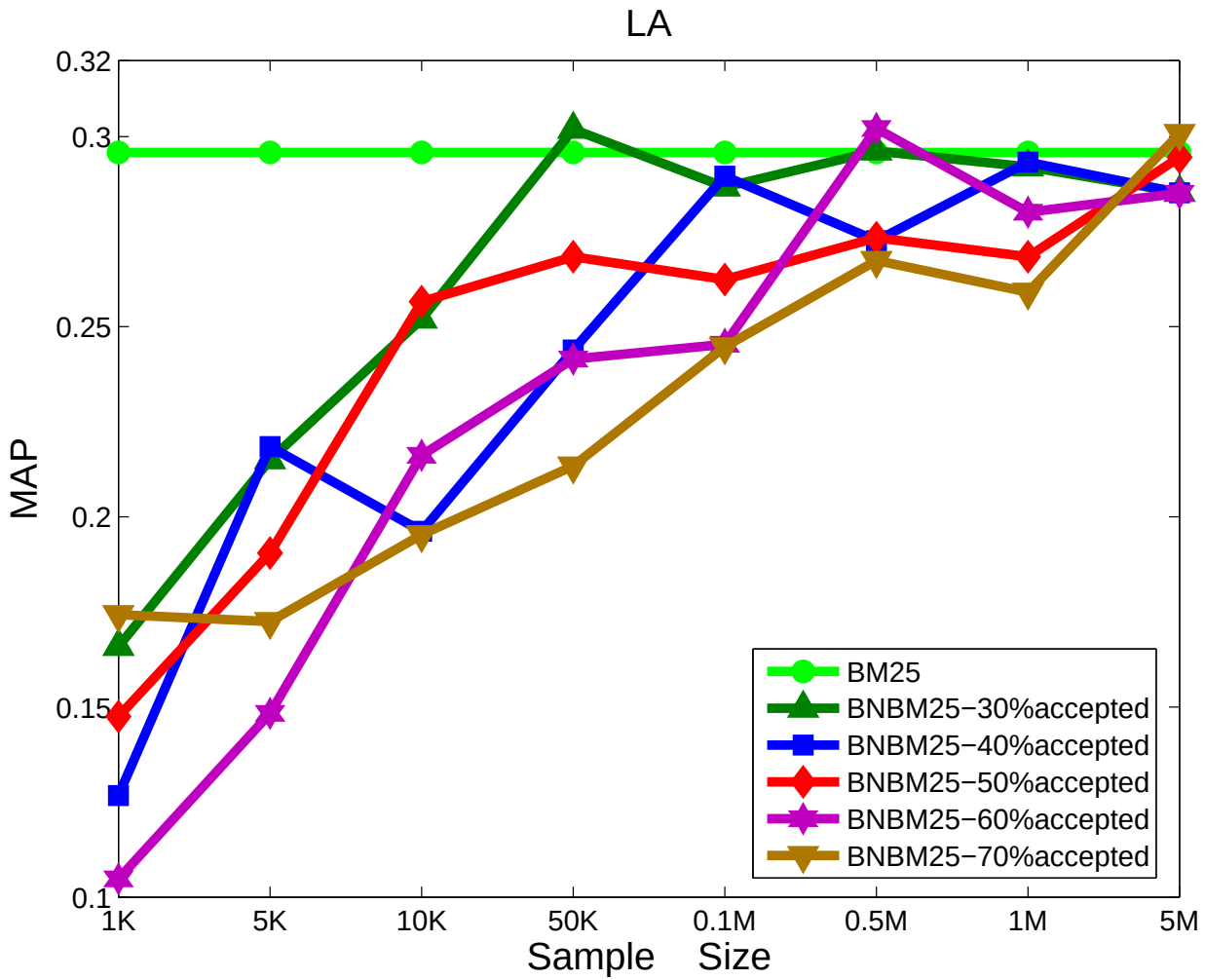
Experimental Results: Comparisons of BNBM25 and BM25 (Cont'd)



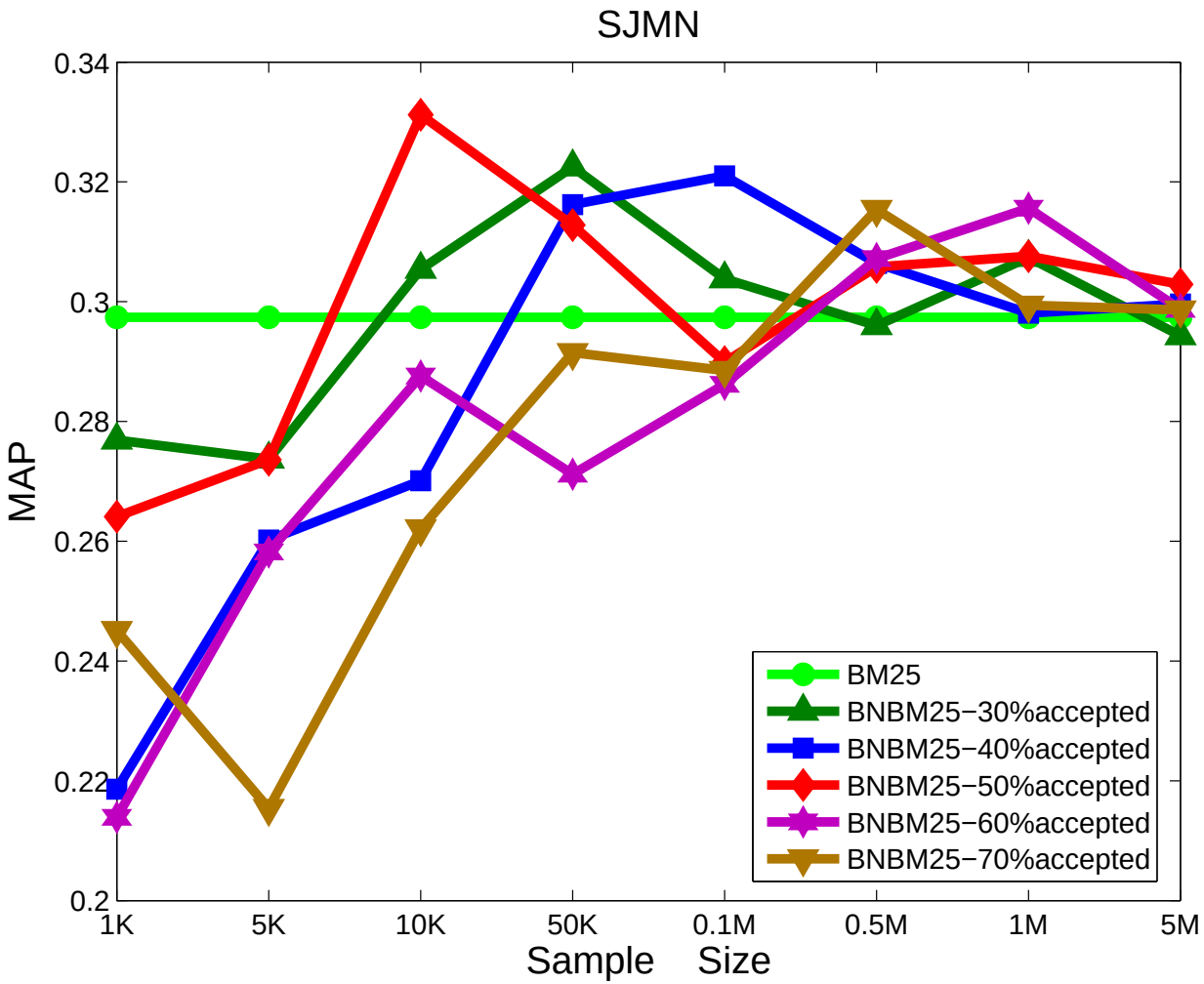
Experimental Results: Comparisons of BNBM25 and BM25 (Cont'd)



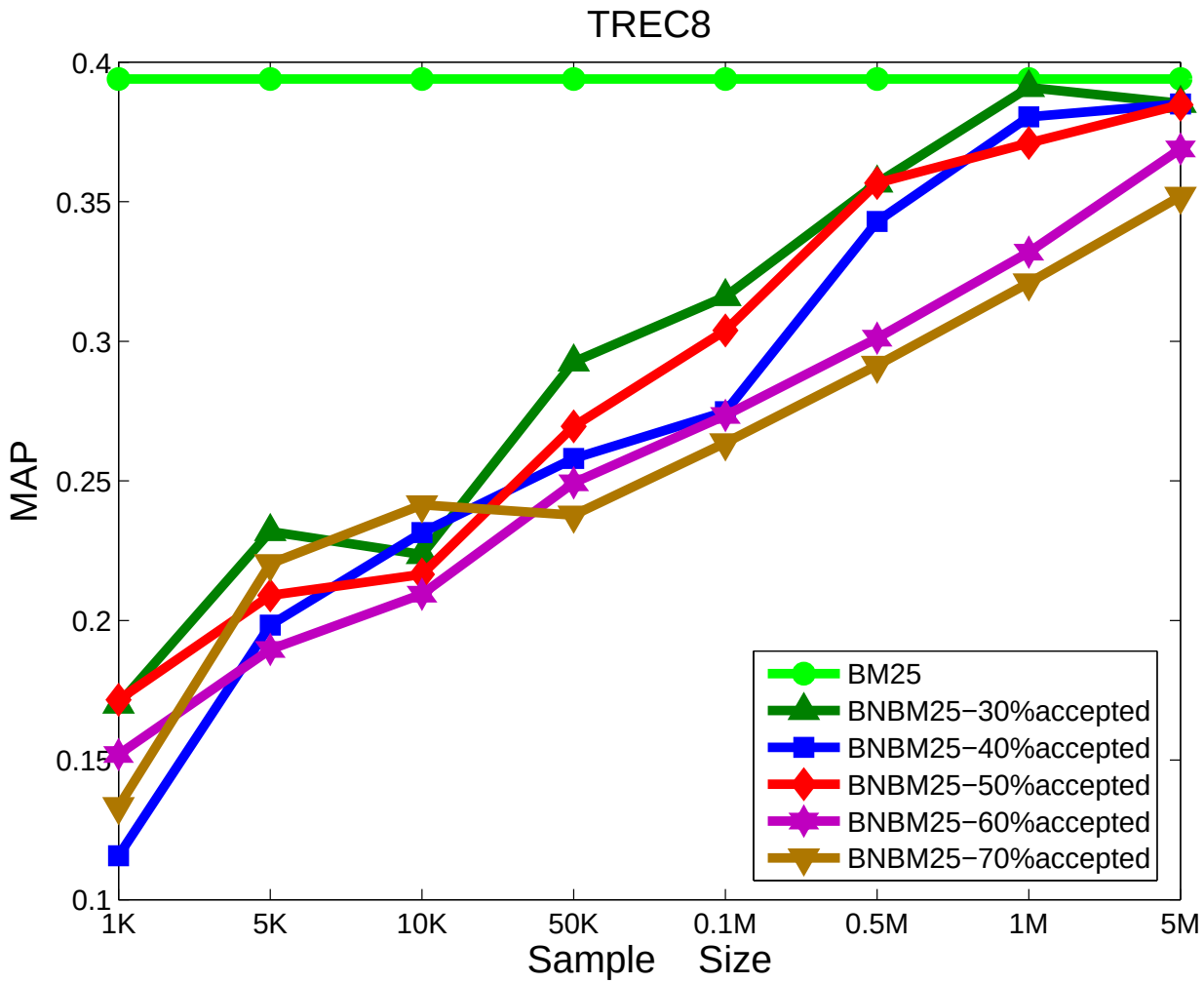
Experimental Results: Comparisons of BNBM25 and BM25 (Cont'd)



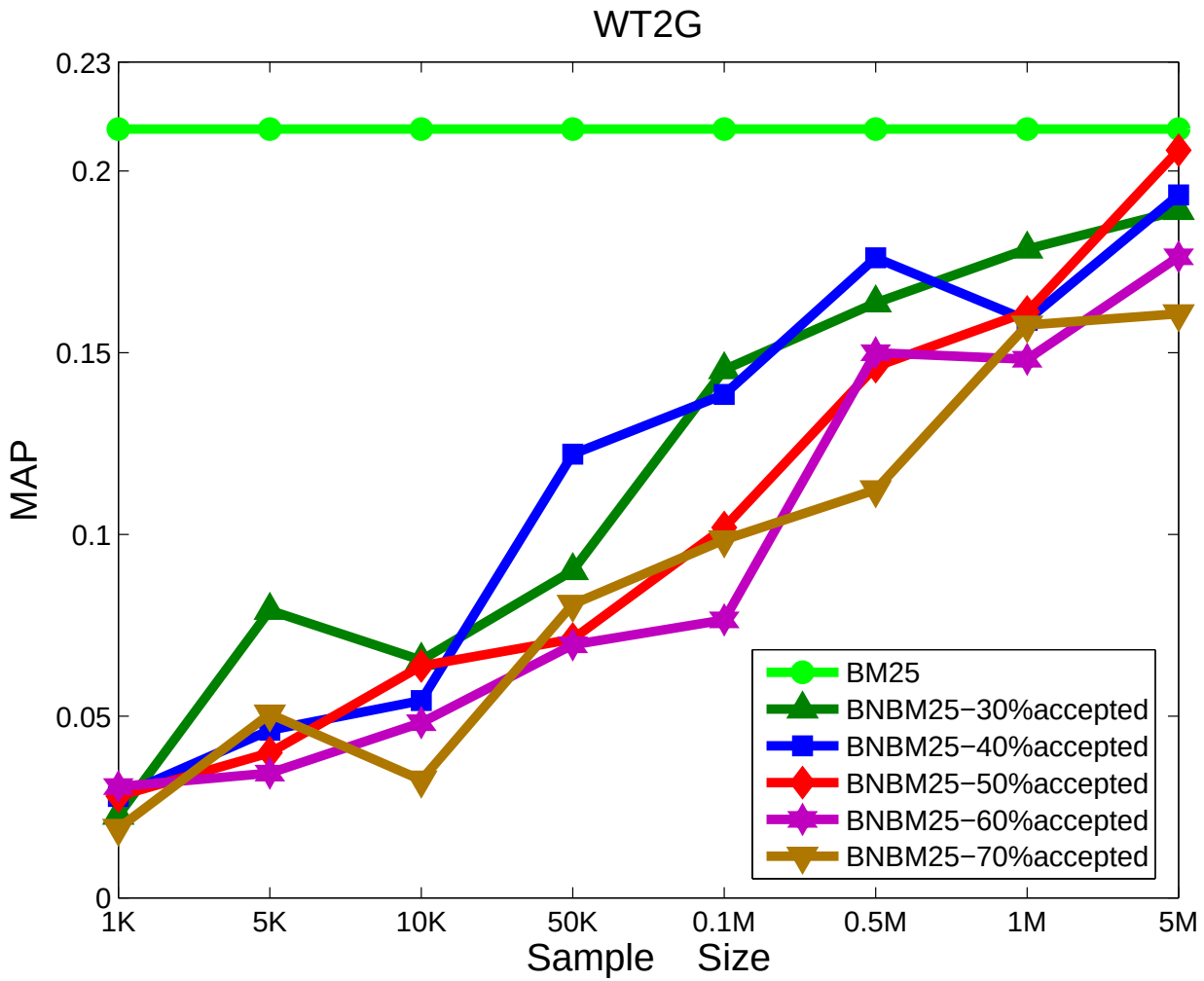
Experimental Results: Comparisons of BNBM25 and BM25 (Cont'd)



Experimental Results: Comparisons of BNBM25 and BM25 (Cont'd)



Experimental Results: Comparisons of BNBM25 and BM25 (Cont'd)



Remarks

- Document ranking in IR is transformed into the problems of estimating posterior probabilities through stochastic sampling on a BN designed for IR.
- The framework establishes a cross-bridge connection between probabilistic ranking in traditional information retrieval models and probabilistic inference in Bayesian Networks.
- Experiments show that performance of BNBM25 is at the least comparable to the one of baseline BM25.
- Further refinements on methods calculating the numerical score of a sample can, we believe, lead directly to improved performance of BN-based models;

Remarks (Cont'd)

- Most existing probability theory-based models in IR can actually be adopted in this BN-based framework;
- BN-based models offer new opportunities for getting deeper understanding of existing term weighting models, and for building up better future term weighting models;
- From these BN-based models, document ranking can be further conveyed into a world more expressive in representational power, that is, a world of
  - first-order objects in terms of documents and words;
  - relationships between these objects in view of relevancy, novelty, and diversity; and
  - uncertainties of knowledge about these relationships.