Bayesian Networks

CSE 576: Topics in Natural Language Processing

Dr. Vivek Gupta

Spring 2025

02/03/2025

Recap: N-grams

📌 **Why Probability in NLP?**

- Used for **predicting next words**, resolving **ambiguity** in text. Example: **"I ate a berry!"** is more likely than **"Eye eight uh Jerry"**.

📌 **N-Gram Language Models:**

- Estimate the probability of each word given its **prior context**.
- **Markov Assumption**: Future state depends only on the **previous N-1 states**.
- Unigram, bigrams, trigrams etc.

📌 **Evaluation of Language Models**

- **Perplexity**: Measures how well a model fits test data.
- **Lower perplexity = better language model**.

Recap: N-grams

📌 **Challenges with N-Gram Models**

- **Data sparsity**: MLE assigns **zero probability** to unseen words.
- **Smoothing techniques**: Laplace, Good-Turing, Backoff, Kneser-Ney.
- **Long-distance dependencies**: N-grams struggle to model **syntactic and semantic dependencies**.

📌 **Takeaway:**

- N-grams work **well for local dependencies** but fail for **longer contexts**.
- **We need a better way to model probabilities beyond local context!**
- 🔜 **Enter Bayesian Networks!**

Why Do We Need Bayesian Networks? - Where N-Grams Fall Short

📌 **Limitations of N-Grams**

🚧 **Fixed-size context**: **Cannot** model dependencies across **long distances**.

🚧 **Ignores uncertainty**: Doesn't factor in **external knowledge**.

📌 **Example: Long-Distance Dependencies**

💬 Sentence: **"The computer that crashed yesterday was running an update."**

- **N-grams** might incorrectly predict **"The computer that crashed yesterday *were* running an update."**

- **Why?** "Was" depends on "computer," but the word **"yesterday"** separates them.
  ✅ **Bayesian Networks can capture dependencies between distant words!**

📌 **Example: Word Sense Disambiguation (external knowledge)**

💬 Sentence 1: **"The bank approved my loan."**

💬 Sentence 2: **"I sat by the river bank."**

- **N-grams** only predict based on **previous words**, ignoring meaning.
- ✅ **Bayesian Networks can infer the correct sense of "bank" using context!**

Why Do We Need Bayesian Networks? - Where N-Grams Fall Short

Example: Handling Noisy Input (Speech Recognition & OCR Errors)

📌 **Why N-Grams Struggle with Noisy Data**

- N-grams **assume clean input** and struggle to handle **spelling variations, typos, or speech recognition errors**.
- They only consider **local** word probabilities without **higher-level reasoning**.

📌 **Example: Speech Recognition Failure**

🎤 **User says:** *"I need to book a flight to Nice." (city in France)*
🔄 **Speech recognition system transcribes:** *"I need to book a flight to niece."*
❌ **N-gram model:** "Niece" is more probable than "Nice" in general, so it **chooses the wrong word**.
✅ **Bayesian Networks:**

- Uses **contextual dependencies** (e.g., "flight" → likely a city, not a relative).
- Adjusts **probabilities dynamically** based on the sentence structure.
- Predicts **"Nice" as the correct word** given the **travel-related context.**

Why Do We Need Bayesian Networks? - Where N-Grams Fall Short

📌 **Example: Optical Character Recognition (OCR) Errors**

📝 **Scanned document says:** *"He loves reading b00ks."*

🔄 **OCR misreads:** *"He loves reading looks."*

❌ **N-grams:** "Looks" is more common than "books," so it **accepts the incorrect word**.

✅ **Bayesian Networks:**

- Uses **semantic dependencies** to check if the corrected word **fits the sentence context**.
- Determines that **"books" is more likely than "looks"** after "**reading**."

Why Bayesian Networks Work

🔗 **Graph-Based Representation:**

- Unlike N-grams, which use **linear dependencies**,
- Bayesian Networks use a **Directed Acyclic Graph (DAG)** to capture relationships between variables.
- Each **node** represents a **random variable** (e.g., word, symptom, POS tag).
- Each **edge** represents a **probabilistic dependency** (e.g., "flu" → "fever").

📊 **Models Joint Probability Efficiently:**

- Bayesian Networks **factorize** the **full joint probability distribution**, reducing computation.
- Instead of storing massive probability tables, they break it down into **conditional probability tables (CPTs)**.

💡 **Captures Conditional Dependencies:**

- **Context-aware predictions** → Takes into account multiple factors, unlike N-grams, which rely only on **fixed-length history**.
- **Example:** The probability of "bank" depends on whether "river" or "loan" is in the sentence.

Why Bayesian Networks Work

🔄 **Efficient Inference & Probabilistic Reasoning:**

- **Uses Bayes' Rule** to update **beliefs dynamically**.
- Works well with **missing or uncertain data** (e.g., speech recognition errors, noisy text).

🚀 **Applications in NLP & Beyond:**

- **Word Sense Disambiguation, POS Tagging, Spelling Correction, Speech Recognition, Medical Diagnosis, and Fraud Detection.**

Formal Definition

A Bayesian Network (also called belief network or bayesian belief network) is a **Directed Acyclic Graph (DAG)**, where:

- Each **node** represents a random variable **X** (can be discrete or continuous).
- Each **directed edge** represents a probabilistic dependency **P(X | Parents(X))**.
- The **joint probability** of all variables **factorizes** as:

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | Parents(X_i))$$

- **DAG structure ensures** no cycles, making inference tractable.

The Alarm Example

Suppose you **hear a house alarm go off**. What are the possible causes?

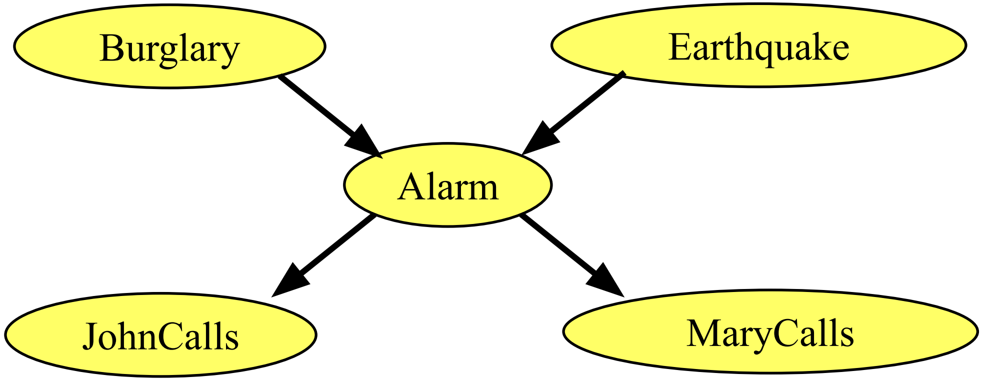
- The alarm could be triggered by a **burglary** or an **earthquake**.
- Your **neighbors (John & Mary)** might call you if they hear the alarm.

How do we quantify the probability of these events?

Instead of assuming **everything is equally likely**, we model **dependencies** between variables.

We represent this scenario as a **Bayesian Network DAG:**

- **Nodes (Random Variables)**



- **B** = Burglary occurred (True/False)
- **E** = Earthquake occurred (True/False)
- **A** = Alarm went off (True/False)
- **J** = John called the police (True/False)
- **M** = Mary called the police (True/False)

- **Edges (Dependencies)**

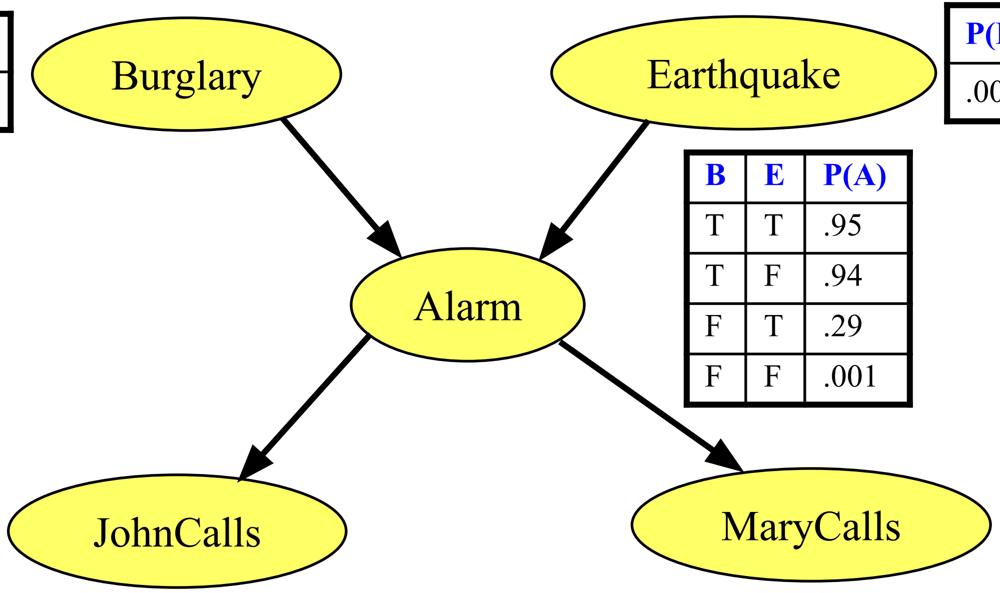
- **B & E** → **A** (The alarm depends on burglary or earthquake)
- **A** → **J & M** (John and Mary call based on the alarm)

- A **Bayesian Network** allows us to compute:
    - **P(Burglary | Alarm went off?)**
    - **P(John Calls | No Alarm?)**
- Instead of a **flat joint probability table** (which would be huge), the **Bayesian Network factorizes** the relationships, i.e. use **conditional probability** making inference efficient!

Conditional Probability Tables

Each node is assigned a Conditional Probability Table (CPT) that specifies probabilities based on parent nodes.

- Roots(sources) of the DAG that have no parents are given prior probabilities.



P(B)
.001

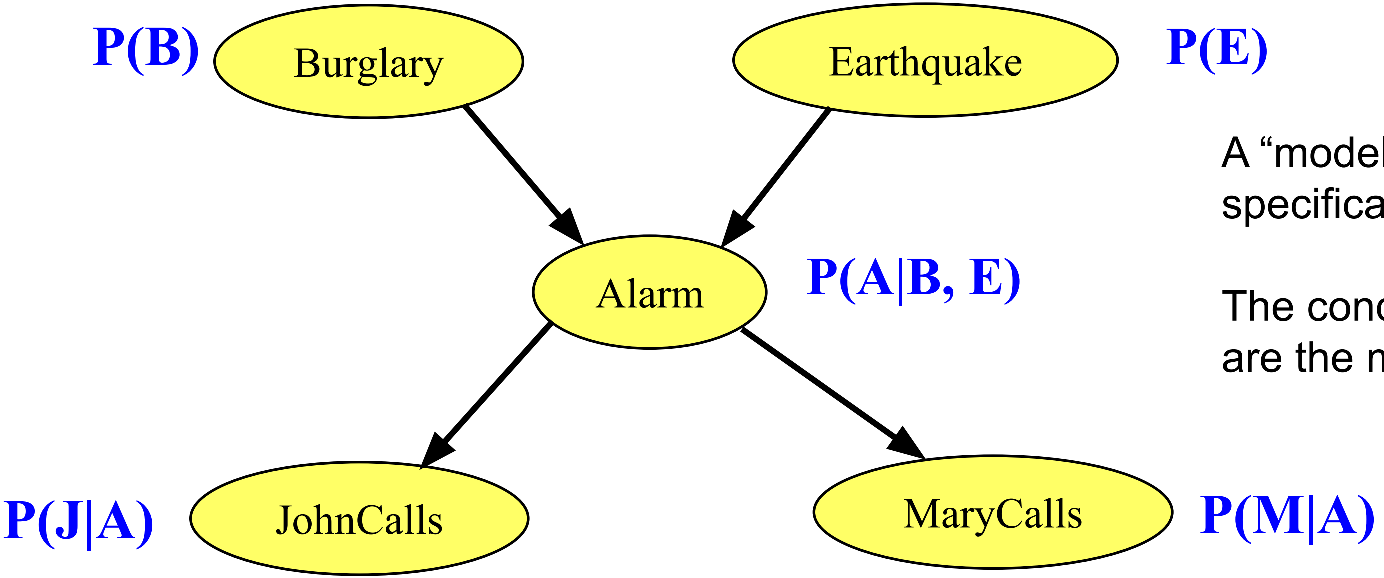
P(E)
.002

B	E	P(A)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J)
T	.90
F	.05

A	P(M)
T	.70
F	.01

Conditional Probability Tables

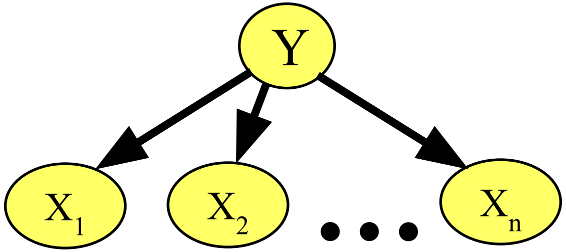


A “model” is a complete specification of the dependencies.

The conditional probability tables are the model parameters.

Naïve Bayes as a Bayes Net

- Naïve Bayes is a simple Bayes Net



- Priors P(Y) and conditionals P(X_i|Y) for Naïve Bayes provide CPTs for the network.

Conditional Probability Tables

- Without Bayesian Networks, we would need the **full joint probability table**, which requires storing probabilities for **all possible variable combinations**.

- When specifying a **Conditional Probability Table (CPT)** for a **Boolean variable**, we only need to provide the probability of one outcome (e.g., `True`) because the probability of the other outcome (`False`) is **implicitly determined**.

  P(Mary call | Alarm off) = 0.7 → P(Marry call | ¬ Alarm off) = 0.3

- In the example shown before,
  - Without factoring, we need $2^5 - 1 = 31$ parameters.
  - Using CPTs, we only need **10 parameters**, a **significant reduction**.
- Number of parameters in the CPT for a node is **exponential** in the number of parents (fan-in problem) → $2^{(\text{parents nodes})}$

Conditional Probability Tables

How to calculate these probabilities?

- Estimate using data, using the Maximum Likelihood Estimation ($P(Y|X) = ?$)

$P(Y|X) = P(Y \cap X)/P(X) \rightarrow P(Y \cap X) = P(Y|X) \times P(X)$

$P(X|Y) = P(Y \cap X)/P(Y) \rightarrow P(Y \cap X) = P(X|Y) \times P(Y)$

$\rightarrow P(Y|X) = P(X/Y) \times P(Y) \,/\, P(X)$

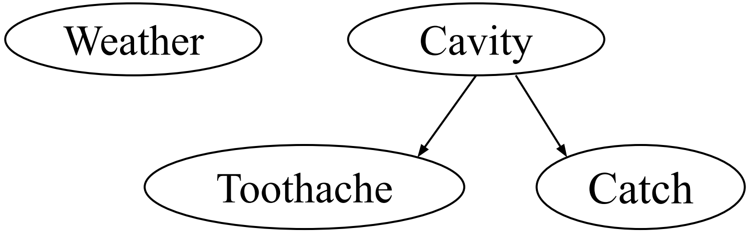
Posterior

Prior's

- What if we don't have Data? (Domain Knowledge Approach)
  - If data is unavailable, experts assign **reasonable probability estimates** based on experience.
  - Example: If earthquakes are rare and alarms are sensitive, we may **manually define** probabilities.

$$P(A = True|B, E) = \frac{\text{Count of A=True when B,E occur}}{\text{Total count of B,E}}$$

Another example - Bayesian Network



P(A|B,C) = P(A|C)
I(ToothAche,Catch|Cavity)

- Weather is independent of the other variables,
  - I(Weather,Cavity)
  - or P(Weather) = P(Weather|Cavity) = P(Weather|Catch) = P(Weather|Toothache)
- Toothache and Catch are conditionally independent given Cavity
  - I(Toothache,Catch|Cavity) meaning
  - P(Toothache|Catch,Cavity) = P(Toothache|Cavity)

Full Joint Distribution

- We will use the following abbreviations:
  - P(x_1, …, x_n) for P($X_1 = x_1 \wedge \ldots \wedge X_n = x_n$)
  - *parents*(X_i) for the values of the parents of X_i

- From the Bayes net, we can calculate:

$$P(x_1,\ldots,x_n) = \prod_{i=1}^{n} P(x_i \mid parents(X_i))$$

Full Joint Distribution

$$P(x_1,\ldots,x_n)$$

$$= P(x_n \mid x_{n-1},\ldots,x_1)P(x_{n-1},\ldots,x_1)$$

$$= P(x_n \mid x_{n-1},\ldots,x_1)P(x_{n-1} \mid x_{n-2},\ldots,x_1)P(x_{n-2},\ldots,x_1)$$

$$= P(x_n \mid x_{n-1},\ldots,x_1)P(x_{n-1} \mid x_{n-2},\ldots,x_1)\ldots P(x_2 \mid x_1)P(x_1)$$

$$= \prod_{i=1}^{n} P(x_i \mid x_{i-1},\ldots,x_1)$$

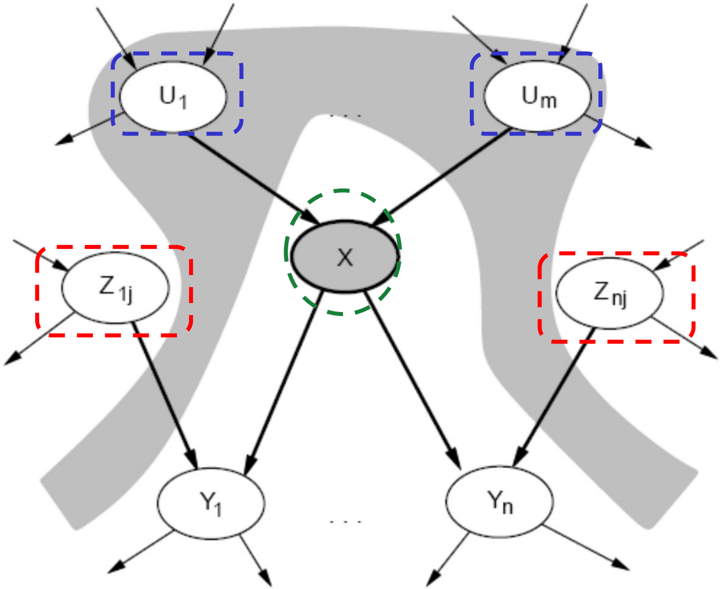
using Independence

$$= \prod_{i=1}^{n} P(x_i \mid parents(x_i))$$

Conditional Independence

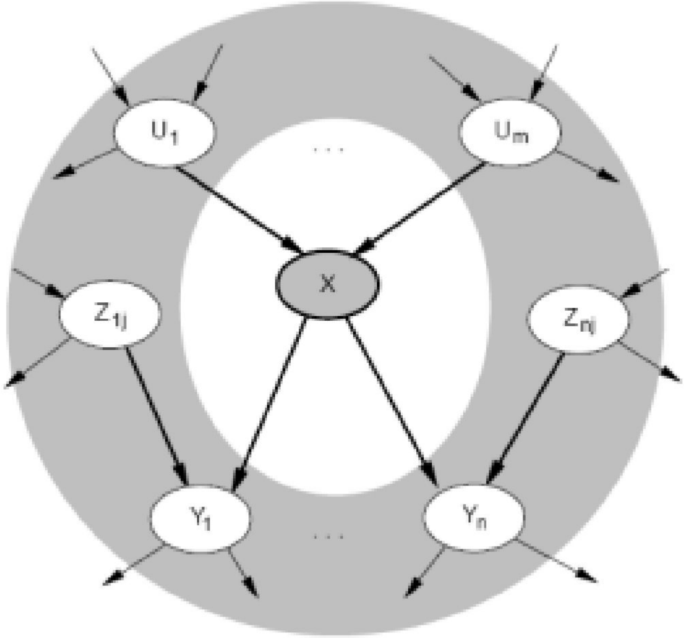
We can look at the actual graph structure and determine conditional independence relationships.

A node (X) is conditionally independent of its non-descendants (Z_{1j}, Z_{nj}), given its parents (U_1, U_m).

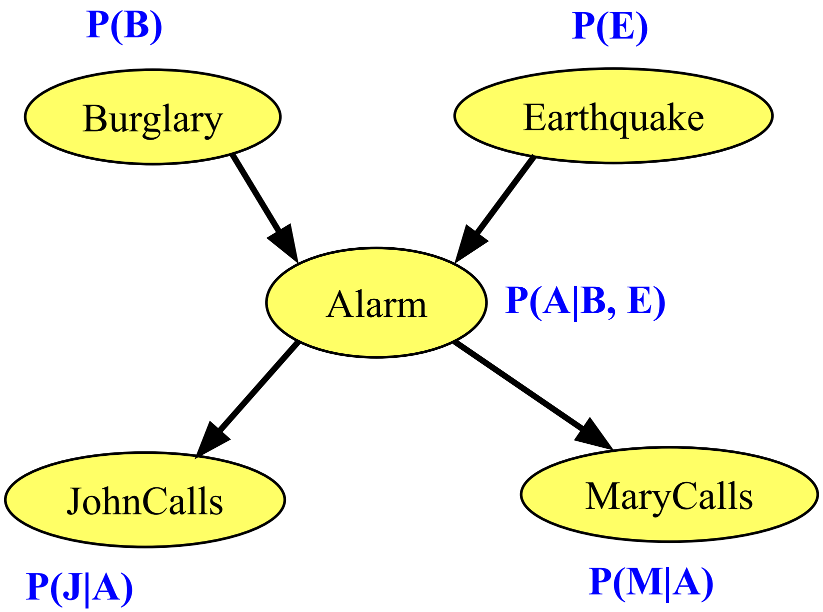


Conditional Independence

Equivalently, a node (*X*) is conditionally independent of all other nodes in the network, given its parents (U_1, U_m), children (Y_1, Y_n), and children's parents (Z_{1j},Z_{nj}) – that is, given its Markov blanket



Independence $\neq$ Conditional Independence



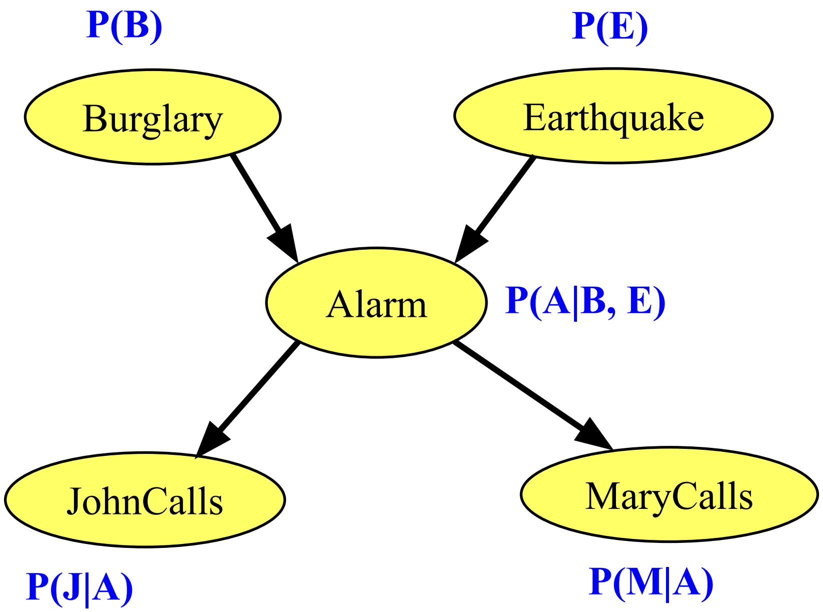
B and E are **independent**:

$$P(B|E) = P(B)$$

B and E are **not conditionally independent given A**:

$$P(B|E, A) \neq P(B|A)$$

Conditional Independence ≠ Independence



J and M are **<u>conditionally independent given A:</u>**

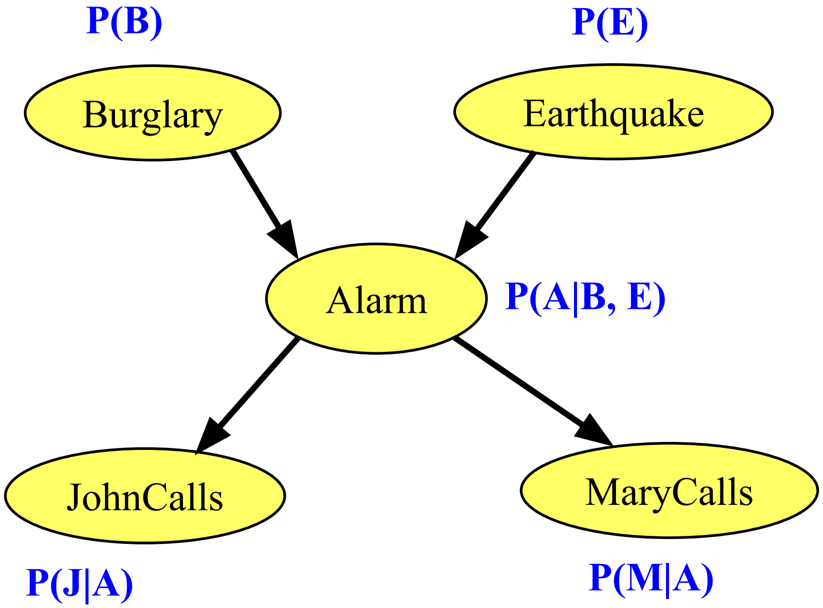
$$P(J|A, M) = P(J|A)$$

$$P(M|A, J) = P(M|A)$$

J and M are **<u>not independent</u>**!

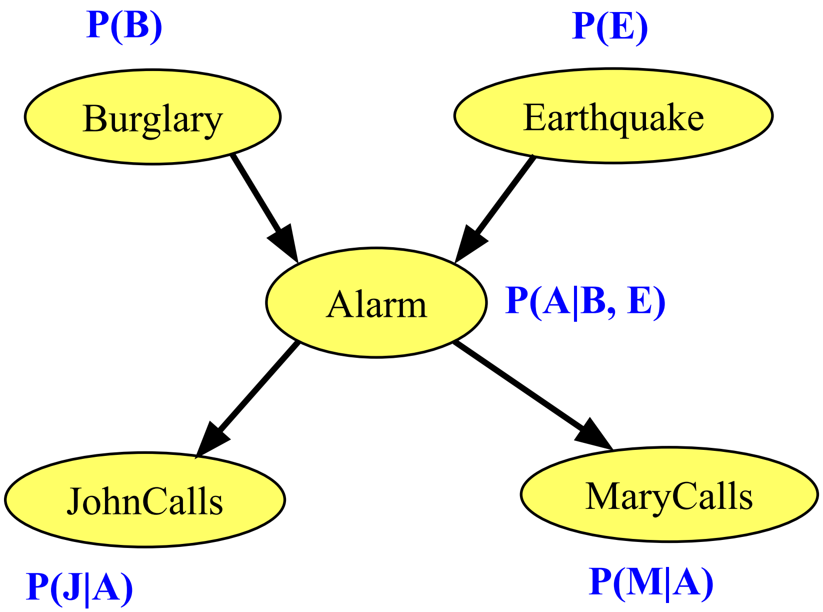
$$P(J|M) \neq P(J)$$

Conditional Independence ≠ Independence



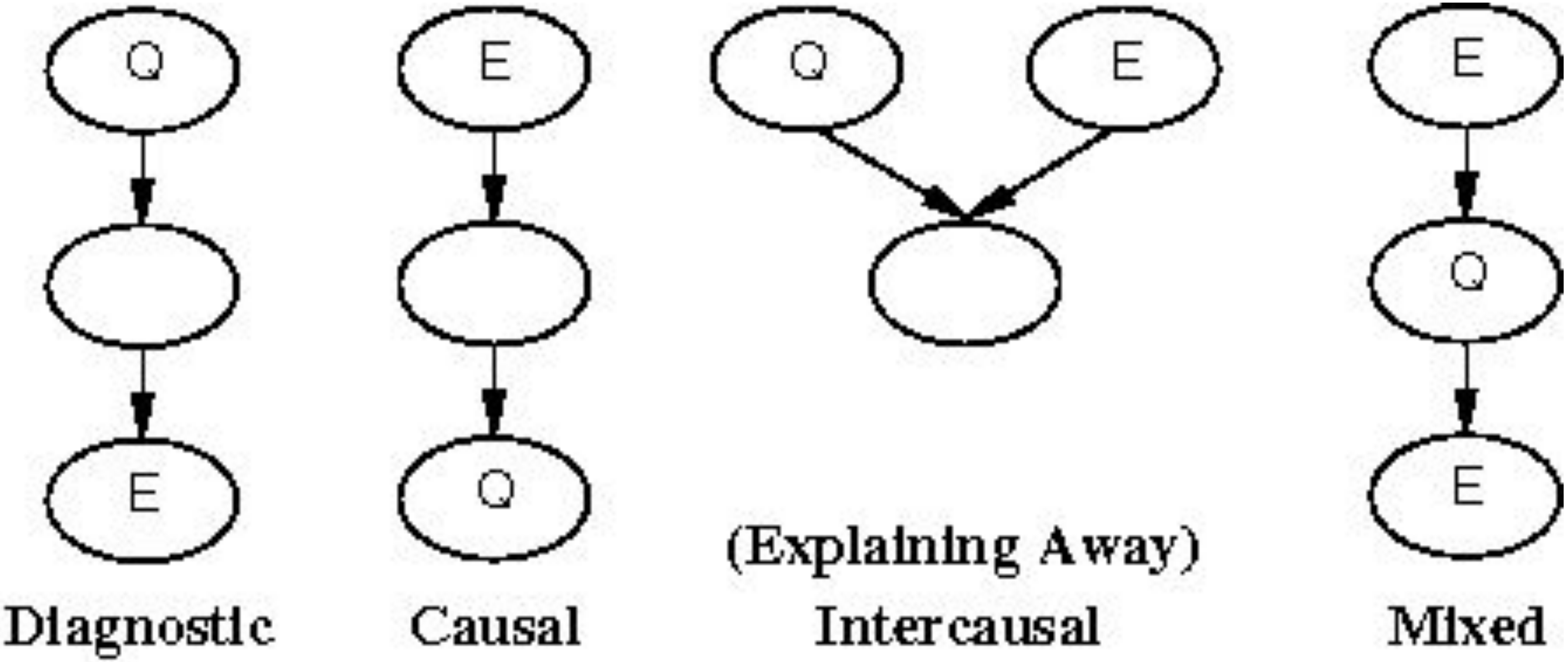
- B and E (no common ancestor, common descendant A):
  - Independent
  - Not conditionally independent given A
- J and M (common ancestor A, no common descendant):
  - Not independent
  - Conditionally independent given A
- B and M (B is the ancestor of M):
  - Not independent
  - Conditionally independent given A

Bayes Net Inference



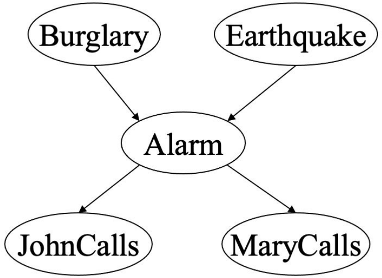
- Given known values for some **evidence variables**, determine the **posterior probability** of some **query variables**.
- **Example:** Given that John calls, what is the probability that there is a Burglary? **P(B|J)**

Types of Inference



Solve - Diagnostic (evidential, abductive)

From effect to cause



P(B)
.001

P(E)
.002

B	E	P(A)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J)
T	.90
F	.05

A	P(M)
T	.70
F	.01

P(B|J)

Given,

$P(B) = P(B|E)$, $P(E) = P(E|B)$,

$P(A|B,E)$,

$P(J|A) = P(J|MA)$, $P(M|A) = P(M|JA)$

$$P(B|J) = P(J|B) \times P(B) / P(J)$$

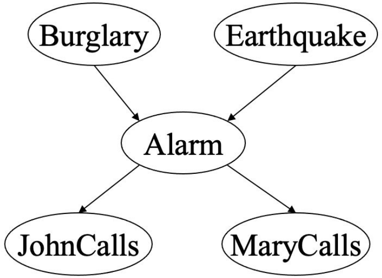
$\rightarrow P(J|B) = \sum_{(A,E)} P(J|A) \times P(A|B,E) \times P(E)$

$\rightarrow P(J) = \sum_{(A)} P(J|A) \times P(A)$

$\rightarrow P(A) = \sum_{(B,E)} P(A|B,E) \times P(B) \times P(E)$

Solve - Causal (predictive)

From cause to effect



P(J|B)

$$P(J|B) = \sum_{(A,E)} P(J|A) \times P(A|B,E) \times P(E)$$

P(B)
.001

P(E)
.002

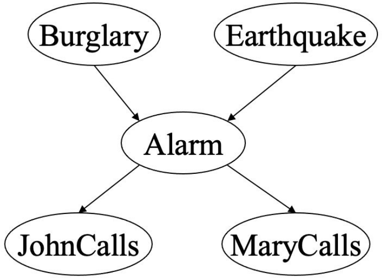
B	E	P(A)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J)
T	.90
F	.05

A	P(M)
T	.70
F	.01

Solve - Intercausal (explain away)

Between causes of a common effect



P(B)
.001

P(E)
.002

B	E	P(A)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J)
T	.90
F	.05

A	P(M)
T	.70
F	.01

P(B|A)

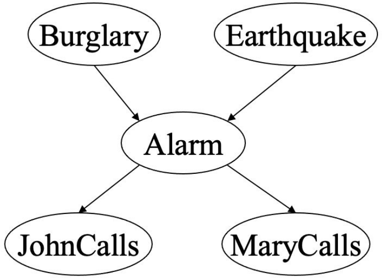
$P(B|A) = P(A|B) \times P(B) / P(A)$

$\rightarrow P(A) = \sum_{(B,E)} P(A|B,E) \times P(B) \times P(E)$

$\rightarrow P(A|B) = \sum_{(E)} P(A|B,E) \times P(E)$

Solve - Mixed

Two or more combination of Diagnostic, Causal, Intercausal



P(B)
.001

P(E)
.002

B	E	P(A)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J)
T	.90
F	.05

A	P(M)
T	.70
F	.01

$$P(A|J \wedge \neg E)$$

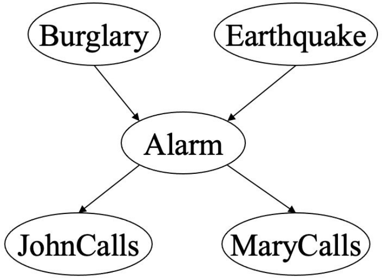
$P(A \mid J,\neg E) = P(J \mid \neg E) \times P(A \mid \neg E) / P(J \mid A)$

$\rightarrow P(A \mid \neg E)=\sum_{(B)} P(A \mid B,\neg E) \times P(B)$

$\rightarrow P(J \mid \neg E)=\sum_{(A)} P(J \mid A) \times P(A \mid \neg E)$

$\rightarrow P(A \mid J,\neg E)=P(J \mid \neg E)\ P(J \mid A) / P(A \mid \neg E)$

Answers



P(B)
.001

P(E)
.002

B	E	P(A)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J)
T	.90
F	.05

A	P(M)
T	.70
F	.01

- $P(B|J) = 0.016$
- $P(J|B) = 0.86$
- $P(B|A) = 0.376$
- $P(A|J \wedge \neg E) = 0.034$