Ira A. Fulton School of Engineering
ARIZONA STATE UNIVERSITY
School of Computing and Informatics

# From NLP to Natural Language Understanding for medical decision making

Chitta Baral

Professor

Arizona State University

# What is Natural Language Understanding?

# Inputs for decision making

- Do various kinds of reasoning with the following kind of knols and knol modules
  - **Facts:** e.g., Various kinds of interactions
  - **General and domain specific rules**: e.g., Rules about pharmacokinetics, how can drugs interact, rules about constructing pathways
  - **General reasoning mechanisms**
    - Explanation, diagnosis, prediction, planning and design, etc.

# Where do we get the “knols” from

- Facts: Databases, Text
- General and domain specific rules: Expert knowledge, Text
- Reasoning Rules and Modules: Given (already known); Develop them.

# NLP to NLU

- NLP: Extract facts from text
  - Automatics Extraction
  - Collaborative development of databases
- NLU: Obtain more general knowledge from the text that can be used together with extracted facts for understanding (i.e., answer various kinds of questions.)

# Extracting Facts from Text

- For example, some of the azole antifungals are inhibitors of both P450 enzymes and P-glycoprotein (Nivoix et al., 2008), whereas rifampicin is an inducer of both CYP3A4 and P-glycoprotein (Katragadda et al., 2005).

# Extracting more general knowledge from text

- While the importance of metabolism in many drug-drug interactions is beyond question, it has become increasingly apparent in recent years that inducers and inhibitors of some of the enzymes of drug metabolism can also affect drug transporter proteins.
- Hence, interaction can sometimes involve drug-metabolizing enzymes, drug transporters, or both.

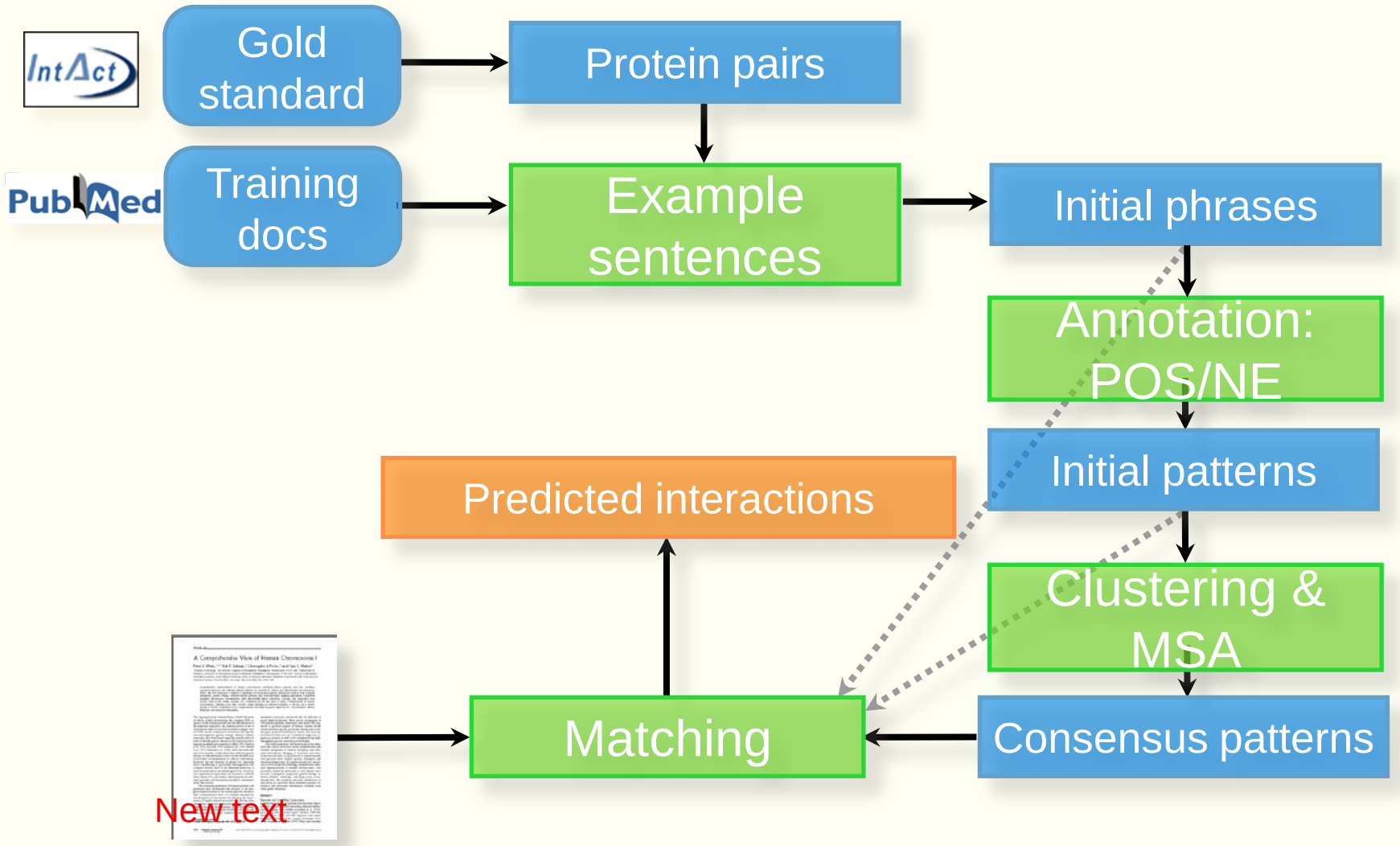
# Outline of the rest



- More general fact extraction: From traditional fact extraction to a query based approach
  - Example of a traditional approach - YAPPIE
  - Generalizing text extraction – querying annotated parse trees
- Examples of decision making
  - How do processes materialize: building pathways
  - Should these drugs be prescribed at the same time? Can they interact?
- Obtaining more general knowledge from text needed for NLU.

# Example of a traditional approach to extract facts from text: protein-protein interactions

# Yappie – Work flow



# Yappie – Initial phrases

- >120,000 snippets that discuss PPI, such as

| |
|---|
| .. P suppressed P .. |
| .. P helps regulate P .. |
| .. P recruits the adapter molecule P .. |
| .. P binding domain of P .. |
| .. P binds directly to the extracellular domain of P .. |
| .. P associates with a novel P dependent kinase, P .. |
| .. while P activation reduces P expression/activation .. |
| .. P was previously found to interact with the KRAB silencing domain of P and with the P .. |

# Yappie – Multiple phrase alignment

Initial phrases:

| | | | | | |
|---|---|---|---|---|---|
| protein | strongly | binds | to | | protein |
| protein | | interacts | with | the | protein |
| protein | never | binds | to | | protein |
| protein | | regulates | | the | protein |
| protein | | inhibits | | a | protein |

Consensus pattern:

| | | | | | |
|---|---|---|---|---|---|
| PROTEIN | {strongly,never} | {binds, .., ..} | {to, with} | {the, a} | PROTEIN |

would exactly match the sentence (part):

| | | | | | |
|---|---|---|---|---|---|
| protein | | binds | to | the | protein |

# Performance: PPI extraction

- #4 system in **BioCreative 2** for protein-protein interactions (**2007**)
- f-measure of 24%, respectively (1st: 30%)
- 20 participants

- #1 system for PPIs in **BioCreative II.5 (2009)**
- 30% f-score (2nd: 23%)
- 15 participants
- >100 submissions overall (multiple configurations per participating team allowed
- Main Person leading this at ASU: Joerg Hakenberg (Now at Roche)

# BioCreative II.5 challenge

- Participated 2 of 3 tasks
  - INT: Interactor normalization task (1st )
  - IPT: Interaction pair task (1st )
- http://www.biocreative.org/news/chapter/biocreative-ii5/
- Main person in our group on this: Joerg Hakenberg

# Outline of the rest



- More general fact extraction: From traditional fact extraction to a query based approach
  - Example of a traditional approach - YAPPIE
  - Generalizing text extraction – querying annotated parse trees
- Examples of decision making
  - How do processes materialize: building pathways
  - Should these drugs be prescribed at the same time? Can they interact?
- Obtaining more general knowledge from text needed for NLU.

# Generalizing text extraction: Querying annotated Parse Trees

# Motivation



- Traditional information extraction technique works as a pipeline
  - Perform grammar parsing, named entity recognition, normalization, extraction
- Information extraction is seen as a one-time process
- Common issues in the development of extraction system
  - What if we change our extraction goals?
    - e.g. extract gene-disease associations rather than protein-protein interactions
  - What if we have an improved NER system?
  - Which of the extraction patterns work well?

# What's needed for extraction?

- To minimize reprocessing, we need to store parse trees and semantic information
  - a database is ideal to store information that we need to perform extraction
- Extraction should be seen as generic
- Can we use database queries as information extraction?
  - Hard to express syntactic patterns with SQL
  - We needed a different kind of query language for extraction; parse tree query language (PTQL)

# Parse trees

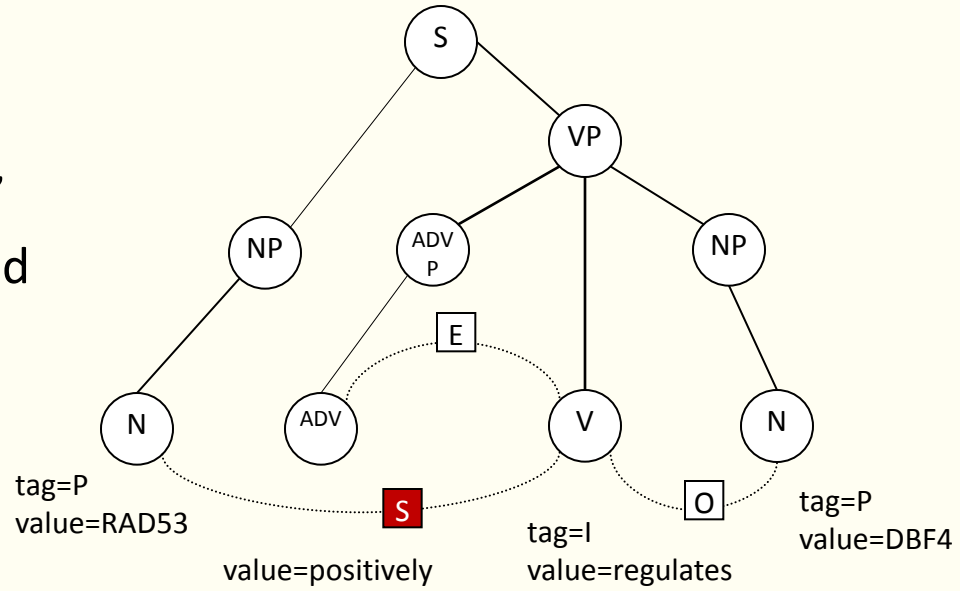


- Stores dependency linkages and constituent trees
- Linkage: shows the dependencies between words in a sentence

```
   +---------S---------+
   |         +----E-----+---O----+
   |         |          |        |
RADB53 positively regulates.v DBF4 .
```

- S: connects subject-noun
- E: verb-modifying adverbs
- O: transitive verbs to direct or indirect objects

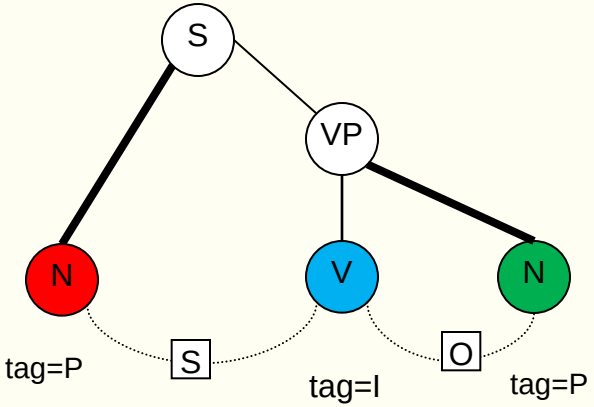
- Constituent trees are represented “vertically”
- Linkages are represented “horizontally”



# PTQL query syntax



- A PTQL query has 4 components in this format
  - *tree pattern* : *link condition* : *proximity condition* : *return expression*
- Tree pattern
  - X{...Y...}: Y is a node in the subtree with X as the root
  - /: parent/child relation in the constituent tree
  - //: ancestor/descendant relation in the constituent tree
  - Example: //S{//N[tag='P']->/VP{/V[tag='I']->//N[tag='P']}}

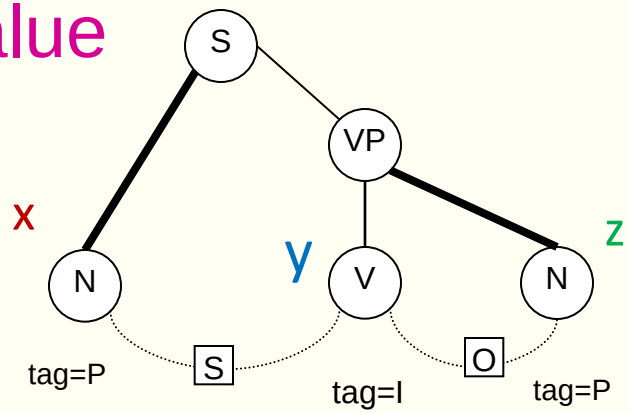


# PTQL query syntax



- To describe horizontal order of nodes:
  - *x* -> *y*: *x* immediately follows *y*
  - *x* => *y*: *x* follows *y*
- Tree pattern : Link condition : : Return expression

//S{//N[tag='P'](x)->/VP{/V[tag='I'](y)->
//N[tag='P'](z)}} : x S y and y O z :: x.value, y.value, z.value



# Other applications of PTQL



- **Feature extraction**
  - Find all MeSH terms and their frequencies among documents that contain recognized gene names.
    - //DOC(x) { //?[tag='GENE'] } : : : count(x.mesh), x.mesh
- **Normalize gene names to species**
  - Find gene-species relations based on some grammatical patterns, such as gene and species occurring in the same noun phrase.
    - //S{//NP{//N[value='human']=>//?[tag='GENE'](x)}} ::: x.value
- **Boosting recall for gene name recognizer**
  - Suppose *"p53"* has been tagged as a gene name in some documents, find *"p53"* such that *"p53"* is not tagged as a gene name.
    - //DOC(x){//STN(y){//?[tag!='GENE' and value='p53']}}::: x.value, y.value

# Outline of the rest



- More general fact extraction: From traditional fact extraction to a query based approach
  - Example of a traditional approach - YAPPIE
  - Generalizing text extraction – querying annotated parse trees
- Examples of decision making
  - How do processes materialize: building pathways
  - Should these drugs be prescribed at the same time? Can they interact?
- Obtaining more general knowledge from text needed for NLU.

# Building pathways; Building reasoning chains

# Building pathways

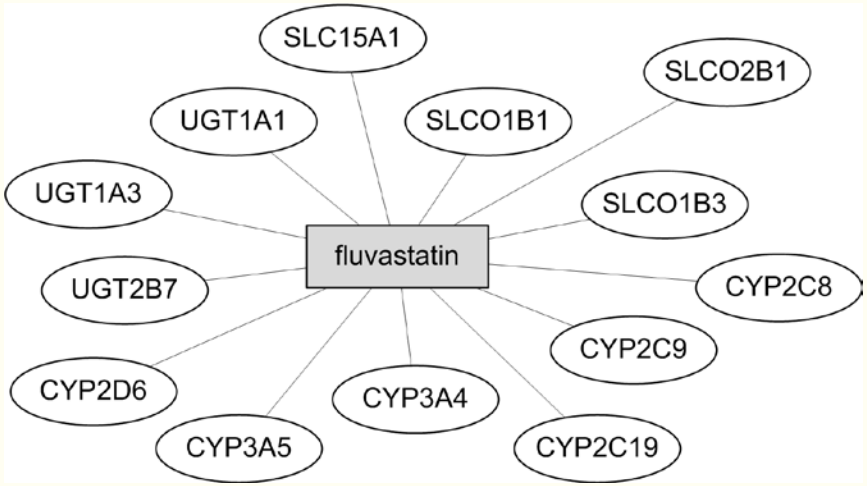


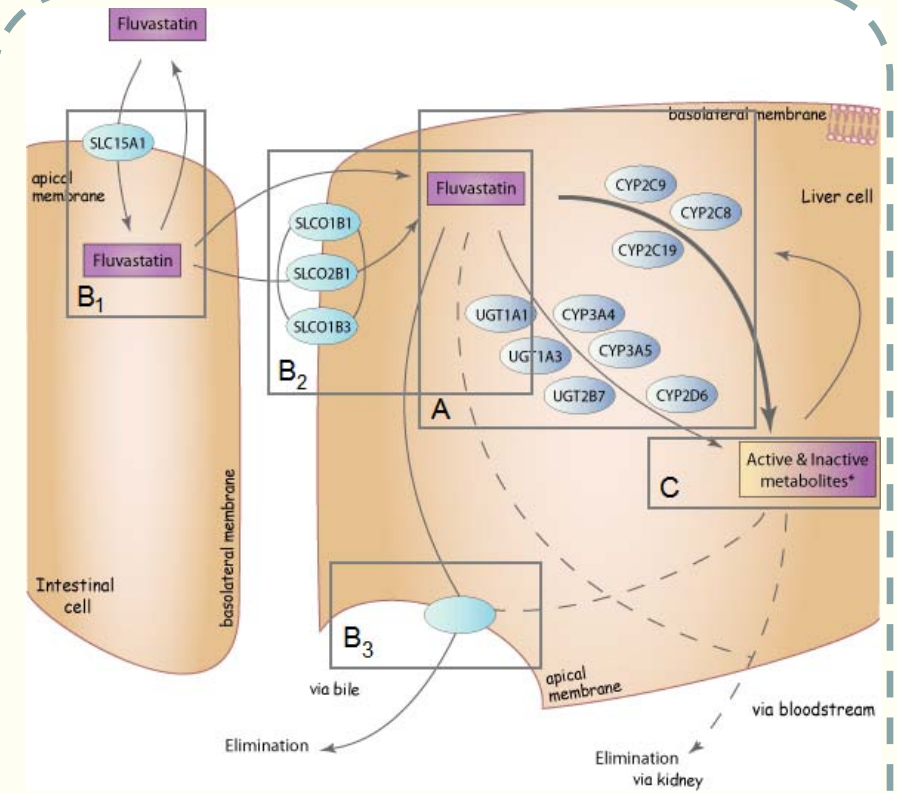
- An important part of understanding or reverse-engineering biological phenomena (disease, phenotype, etc.)
- Connecting the dots !!!
- Building pathways involves
  - Connecting the dots, where the dots are
    - Biological data (such as interactions)
  - But an equally important aspect is
    - Biological Knowledge and
    - Reasoning with that knowledge

# Network vs pathway



Pharmacokinetics represented as a *network*




Pharmacokinetics represented as a *pathway* (from PharmGKB)

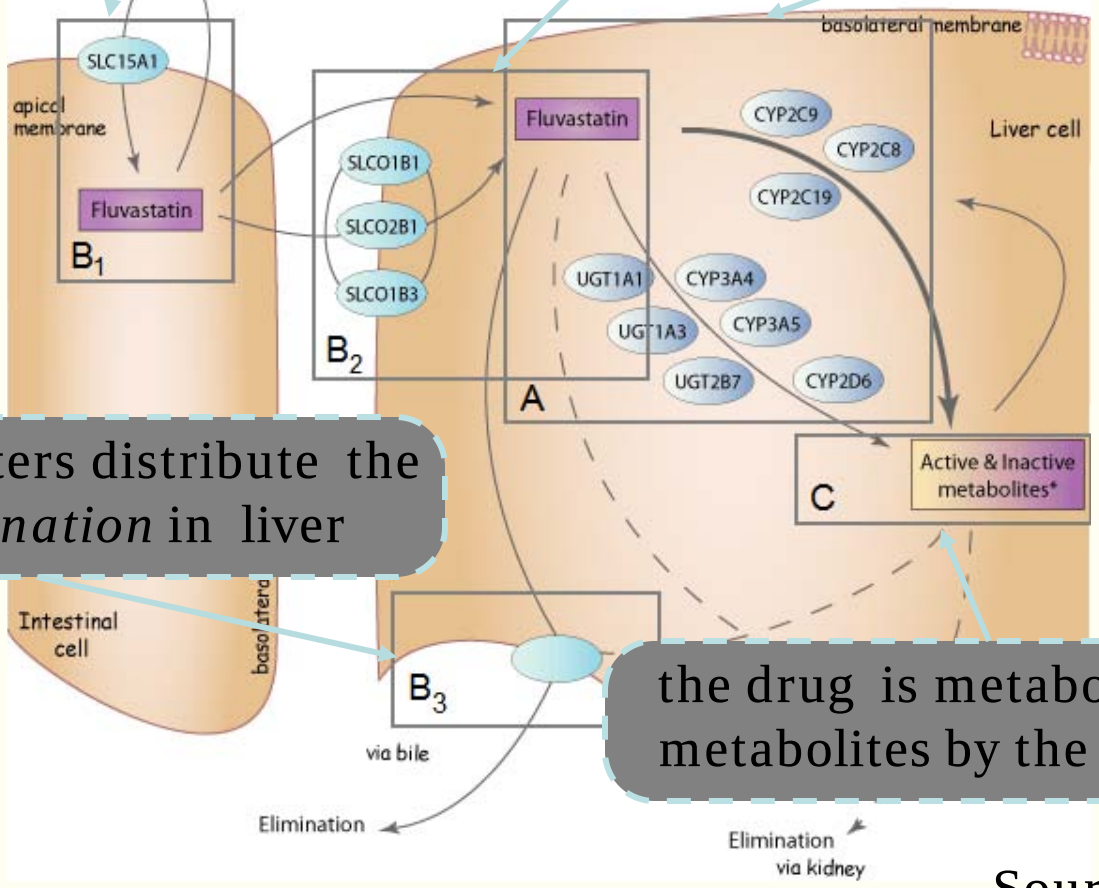
# Pharmacokinetics



drug transporters distribute the drug for *absorption* in

drug transporters distrib for *metabolism* in liver

*metabolism* of the drug by the enzymes




drug transporters distribute the drug for *elimination* in liver

the drug is metabolized to metabolites by the enzymes

Source: PharmGKB

# Pathway synthesis needs

- **Using pharmacokinetic pathway as example**
- **Type of interactions**
  - Knowing that drug *A* interacts with protein *B* is not sufficient
  - We need:
    - Is *A* distributed by transporter *B*?
    - Is *A* metabolized by enzyme *B*?
- **Ordering of interactions**
  - Knowing that drug *A* interacts with transporter *B*, and *A* with enzyme *C* is not sufficient
  - We need: knowledge that captures the fact that *A* has to be distributed by *B* *before* *A* is metabolized by *C*

# Our approach



- Part 1: Data acquisition
  - Fact and interaction extraction from knowledge bases and text
    - Knowledge bases: DrugBank, PharmGKB (drug-gene relations only), Gene Ontology annotations
    - Text: entire collection of Medline abstracts
- Part 2: Automated reasoning using Knowledge
  - Inferences of pathways through reasoning with the extracted interactions
    - Logic rules to capture biological knowledge of pharmacokinetic pathways and order the interactions

# Logic rules



- AnsProlog: for reasoning and representing knowledge
  - Pre- and post-conditions of interactions
  - Timepoints for the logical ordering of interactions
- Sample logic rule describing that the action "metabolize" has to happen before the action "eliminate"

```
o(eliminates(DT,Dr),Loc, T) :-
    h(metabolized(Dr, Loc),T),
    extr_elim(DT,Dr), extr_metabolism(Dr, Loc).
```

from extraction

# Sample logic rules about pharmacokinetics

- Direct effect of an action (post-condition)

```
h(metabolized(D, Loc),T+1) :- o(metabolizes(EN,
  D), Loc, T), not -h(metabolized(D, Loc),T).
```

- Indirect effect of an action (static causal law)

```
-h(is_present(D, Loc), T+1) :- h(eliminated(D,
  Loc),T), metabolism(D, Loc).
```

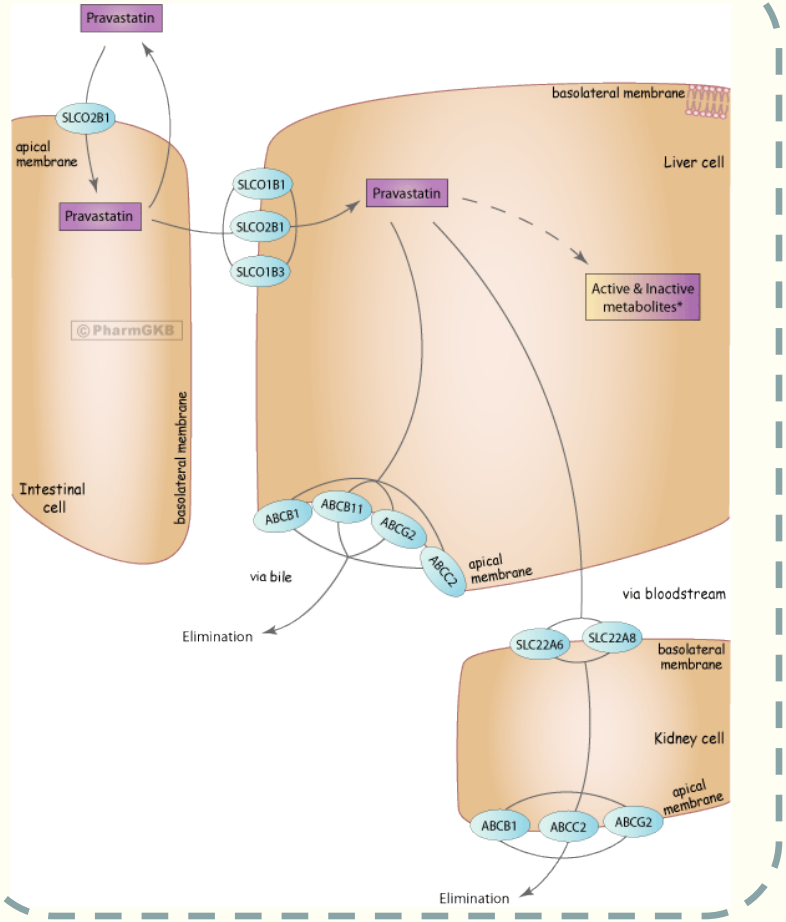
- Constraint – all interactions in intestine must appear before the interactions in liver

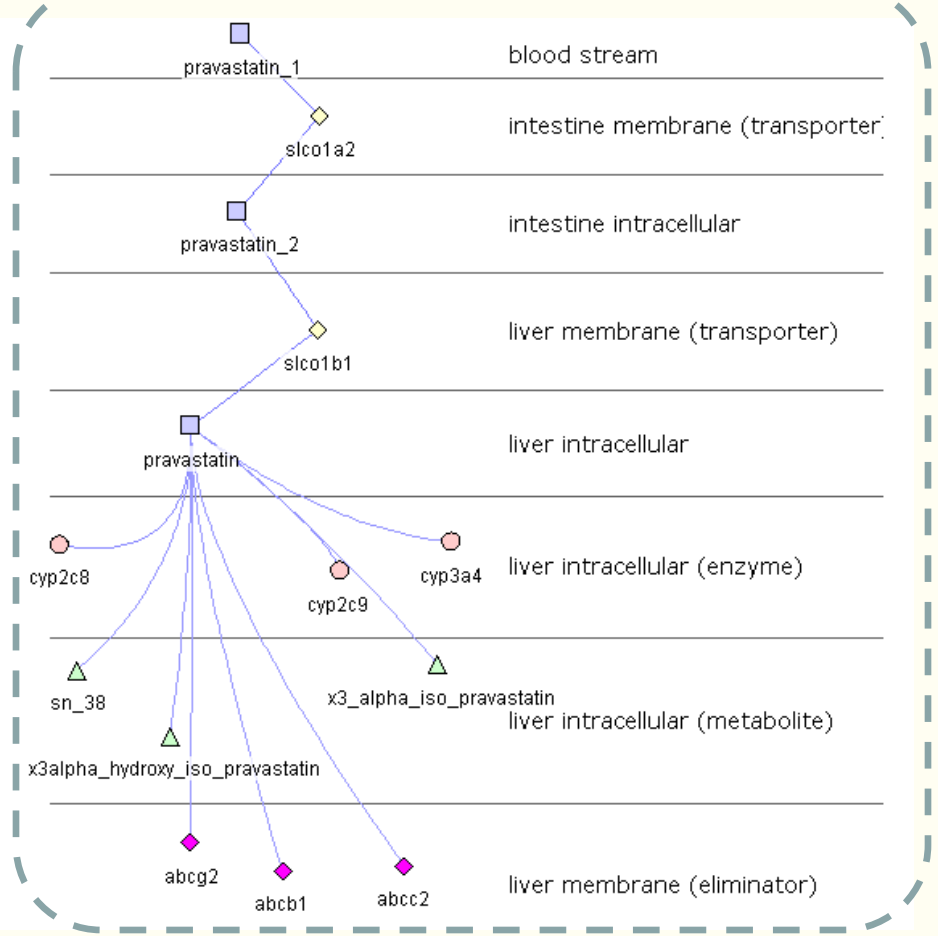
```
:- o(ACT, liver, T), o(ACT1, intestine, T1),
      T <= T1.
```

# System output



- Input: drug name
- Output: models describing each of the pathway steps, represented in Cytoscape Cerebral graphs



# Limitations



- Our synthesized pathways do not capture
  - which enzymes are responsible for the production of a particular drug metabolite
    - Drug-enzyme-metabolite relations can rarely be found within individual sentences
  - transformation of a metabolite to another metabolite through enzymes
    - as suggested by the pathways for phenytoin and tamoxifen
  - close-loop interactions

# Outline of the rest



- More general fact extraction: From traditional fact extraction to a query based approach
  - Example of a traditional approach - YAPPIE
  - Generalizing text extraction – querying annotated parse trees
- Examples of decision making
  - How do processes materialize: building pathways
  - Should these drugs be prescribed at the same time? Can they interact?
- Obtaining more general knowledge from text needed for NLU.

# Drug-drug interactions

# Importance of studying drug-drug interactions

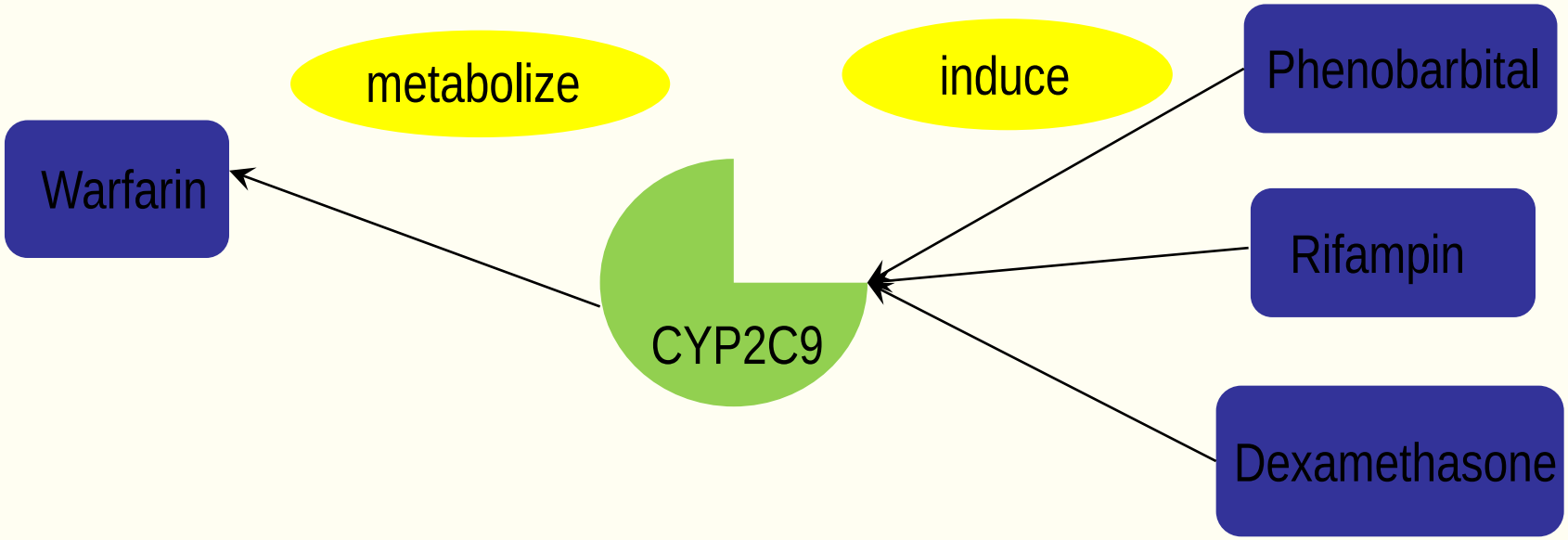
- **Drug design:** Early assessment of a new compound's potential interactions with other drugs can avoid costly investment in the drug discovery process.

- **Drug prescription:** For multi-drug prescription, pharmacokinetic interactions amongst co-administrated drugs may alter the bioavailability of the drugs that can lead to life-threatening side effects for the patients.

# An Example



- S-warfarin, predominantly responsible for the anticoagulation effect, is metabolized mostly by the CYP2C9 enzyme. *[PMID: 19799531]*
- CYP2C9 is subject to induction by rifampin, phenobarbital, and dexamethasone. *[PMID: 19515014]*



# Example cont.



- Consequence:
  - CYP2C9 enzyme activity is increased.
  - Rate of metabolism of warfarin by CYP2C9 is increased.
  - Bioavailability of warfarin is decreased.

# System Overview



| Input | Multi-drug regimen |
|---|---|
| Output | Potential interactions among the input drugs |
| Reasoning task | 1. What drugs can be affected?<br>2. Does the interaction lead to drug toxicity or less efficient?<br>3. Is there any aggregated effect? |

# Data Acquisition



- **Existing Knowledge base of drug-drug interactions:**
  - PharmGKB
  - DrugBank
  - **Pros**: Accurate information about curated drugs.
  - **Cons**: Still remain largely incomplete.

- **We add:**
  - Automated extraction via PTQL

# Extracted Facts



| | <Entity - Entity> | Relation Keywords |
|---|---|---|
| 1 | Drug – Protein | Induce / Inhibit<br>Increase / Decrease |
| 2 | Protein – Drug | Metabolize / Distribute / Eliminate |
| 3 | Protein – Protein | Activate / Suppress<br>Up-regulate / Down-regulate |
| 4 | Protein – Role | Enzyme / Transporter / Eliminator / Transcription Factor |

| | <Entity - Entity> | Relation Keywords <not><br>--- used to filter out false positives |
|---|---|---|
| 1 | Drug – Protein | N_Induce / N_Inhibit<br>N_Increase / N_Decrease |
| 2 | Protein – Drug | N_Metabolize / N_Distribute / N_Eliminate |
| 3 | Protein – Protein | N_Activate / N_Suppress<br>N_Up-regulate / N_Down-regulate |

# Reasoning about *Pairwise* DDI

**Knowledge encoding for enzyme-mediated DDI:**

```
result(Dr1, increases, Dr2) :-
        affects(Dr1, level(P, low)),
        role(P, enzyme),
        relation(P, metabolizes, Dr2).
result(Dr1, decreases, Dr2) :-
        affects(Dr1, level(P, high)),
        role(P, enzyme),
        relation(P, metabolizes, Dr2).
```

Can be obtained from:

1. Drug-Protein relation from direct fact extraction.
2. Drug-Protein + Protein-Protein relation from fact extraction. (see next slide)

- For transporter, etc., the reasoning is similarly encoded.

# Reasoning about *Pairwise* DDI (cont.)

**Knowledge encoding for Transcription Factor (TF)-mediated DDI:**

```
affects(Dr, level(P, high)) :-
        affects(Dr, level(TF, high)),
        role(TF, tf),
        relation(TF, upregulates, P).
affects(Dr, level(P, low)) :-
        affects(Dr, level(TF, high)),
        role(TF, tf),
        relation(TF, downregulates, P).
```

- Then *affects(Dr, level(P, high/low))* will be used to reason for the transcription-factor mediated DDI (*see previous slide*)

# Outline of the rest



- More general fact extraction: From traditional fact extraction to a query based approach
  - Example of a traditional approach - YAPPIE
  - Generalizing text extraction – querying annotated parse trees
- Examples of decision making
  - How do processes materialize: building pathways
  - Should these drugs be prescribed at the same time? Can they interact?
- Obtaining more general knowledge from text needed for NLU.

# Extracting more general knowledge from text needed for NLU

- While the importance of metabolism in many drug-drug interactions is beyond question, it has become increasingly apparent in recent years that inducers and inhibitors of some of the enzymes of drug metabolism can also affect drug transporter proteins.
- … Hence, interaction can sometimes involve drug-metabolizing enzymes, drug transporters, or both.

# The approach



- CCG grammar as syntax and Lambda calculus formulas as semantics of words
- After parsing, the application of lambda calculus expressions as dictated by the parsing gives the meaning of the sentence.
  - The meaning is a formula in a knowledge representation language.
  - Questions also get translated to logical formulas.
- Grammar and Meaning of words can be learned from sample sentences and their meaning.

# Using CCG and Lambda



| Drug | may | decrease | the | amount | of | enzyme | by | binding | to | it |
|---|---|---|---|---|---|---|---|---|---|---|
| $NP$ | $((S\backslash NP)/NP)/NP$ | $NP$ | $NP/NP$ | $NP$ | $(NP\backslash NP)/NP$ | $NP$ | $(S\backslash S)/(S\backslash NP)$ | $(S\backslash NP)/NP$ | $NP/NP$ | $NP$ |
| $NP$ | $((S\backslash NP)/NP)$ | | $NP/NP$ | $NP$ | $(NP\backslash NP)/NP$ | $NP$ | $(S\backslash S)/(S\backslash NP)$ | $(S\backslash NP)/NP$ | $NP$ | |
| $NP$ | $((S\backslash NP)/NP)$ | | $NP$ | | | | $(S\backslash S)/(S\backslash NP)$ | $(S\backslash NP)$ | | |
| $NP$ | $((S\backslash NP)$ | | | | | | $(S\backslash S)$ | | | |
| | $S$ | | | | | | $(S\backslash S)$ | | | |
| | $S$ | | | | | | | | | |

| Drug | may | decrease | the | amount |
|---|---|---|---|---|
| $D$ | $\lambda y.\lambda u.\lambda v.\lambda x.y@v@u : -x@v@u, not\ (y@v@u)$ | $\lambda x.\lambda y.decrease(y,x).$ | $\lambda x.x$ | $amount$ |
| $D$ | $\lambda u.\lambda v.\lambda x.decrease(v,u) : -x@v@u, not\ decrease(v,u).$ | | $\lambda x.x$ | $amount$ |
| $D$ | $\lambda u.\lambda v.\lambda x.decrease(v,u) : -x@v@u, not\ decrease(v,u).$ | | $amount\_E$ | |
| $D$ | $\lambda v.\lambda x.decrease(v, amount\_E) : -x@v@amount\_E, not\ decrease(v, amount\_E).$ | | | |
| | $decrease(D, amount\_E) : -binding(D, amount\_E), not\ decrease(D, amount\_E).$ | | | |

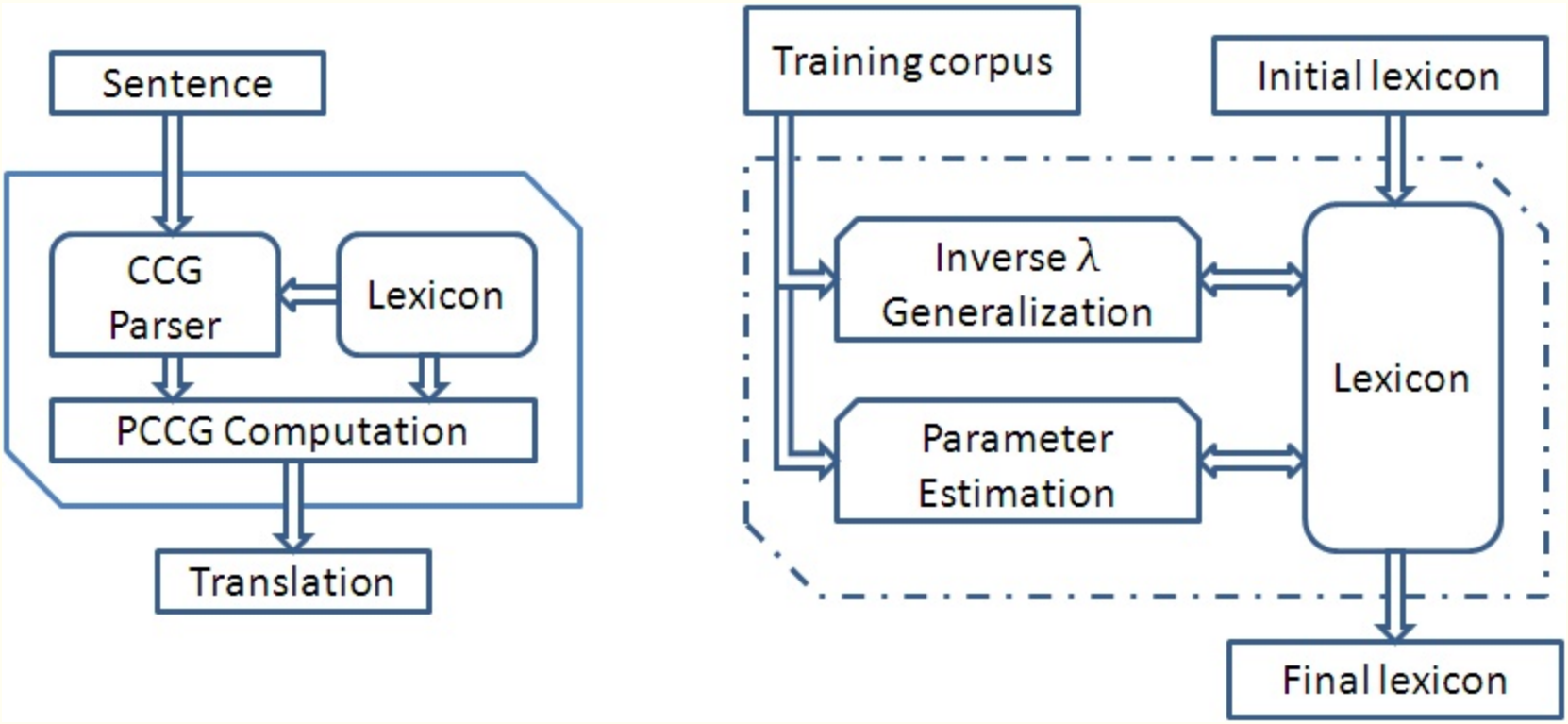
| of | enzyme | by | binding | to | it |
|---|---|---|---|---|---|
| $\lambda x.\lambda y.y\_x$ | $E$ | $\lambda x.\lambda y.y@x$ | $\lambda z.z@\lambda x.\lambda y.binding(y,x)$ | $\lambda x.x$ | $\lambda x.x$ |
| $\lambda x.\lambda y.y\_x$ | $E$ | $\lambda x.\lambda y.y@x$ | $\lambda z.z@\lambda x.\lambda y.binding(y,x)$ | $\lambda x.x$ | |
| | | $\lambda x.\lambda y.y@x$ | $\lambda x.\lambda y.binding(y,x)$ | | |
| | | $\lambda y.y@\lambda x.\lambda y.binding(y,x)$ | | | |
| | | $\lambda y.y@\lambda x.\lambda y.binding(y,x)$ | | | |

| The | inhibition | of | an | enzyme | will | increase | the | level | of | its | substrates. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $NP/NP$ | $NP$ | $(NP\backslash NP)/NP$ | $NP/NP$ | $NP$ | $(S\backslash NP)/NP$ | $NP/NP$ | $NP/NP$ | $NP$ | $(NP\backslash NP)/NP$ | $NP/NP$ | $NP$ |
| $NP$ | | $(NP\backslash NP)/NP$ | $NP$ | | $(S\backslash NP)/NP$ | $NP/NP$ | $NP$ | | $(NP\backslash NP)/NP$ | $NP$ | |
| | | $NP$ | | | $(S\backslash NP)/NP$ | $NP$ | | | | | |
| | | $NP$ | | | $(S\backslash NP)$ | | | | | | |
| | | | | | $S$ | | | | | | |

| The | inhibition | of | an | enzyme | will |
|---|---|---|---|---|---|
| $\lambda x.x$ | $\lambda x.inhibition(x)$ | $\lambda x.\lambda y.y@x$ | $\lambda x.x$ | $E$ | $\lambda x.\lambda y.x : -y.$ |
| $\lambda x.inhibition(x)$ | | $\lambda x.\lambda y.y@x$ | $E$ | | $\lambda x.\lambda y.x : -y.$ |
| | | $inhibition(E)$ | | | $\lambda x.\lambda y.x : -y.$ |
| | | $inhibition(E)$ | | | $\lambda y.increase(level, S, E) : -y.$ |
| | | | | | $increase(level, S, E) : -inhibition(E).$ |

| increase | the | level | of | its | substrates. |
|---|---|---|---|---|---|
| $\lambda x.increase(x)$ | $\lambda x.x$ | $\lambda x.level, x$ | $\lambda x.\lambda y.y@x$ | $\lambda x.x, E$ | $S$ |
| $\lambda x.increase(x)$ | $\lambda x.level, x$ | | $\lambda x.\lambda y.y@x$ | $S, E$ | |
| $\lambda x.increase(x)$ | | | $level, S, E$ | | |
| $increase(level, S, E)$ | | | | | |

# A learning based system to translate English to KR langauges



# The puzzles

```
Puzzle data:
1,2,3,4 and 5 are ranks.
earl, ivana, lucy, philip and tony are names.
earth, fire, metal, water and wood are elements.
cow, dragon, horse, ox and rooster are animals.
```

```
Puzzle clues:

% 1) Tony was the third person to have her fortune told.
% 2) The person with the Lucky Element Wood had their fortune told fifth.
% 3) Earl's lucky element is Fire.
% 4) Earl arrived immediately before the person with the Rooster.
% 5) The person with the Dragon had their fortune told fourth.
% 6) The person with the Ox had their fortune told before the one
who's Lucky Element is Metal.
% 7) Ivana's Lucky Animal is the Horse.
% 8) The person with the Lucky Element Water has the Cow.
% 9) The person with Lucky Element Water did not have their fortune told first.
% 10) The person with Lucky Element Earth had their fortune told exactly
two days after Philip.
```

# Encoding the puzzles in ASP

```
% CLUES

%Tony was the third person to have her fortune told.
:- tuple(I, tony), tuple(J, 3), I!=J.

%The person with the Lucky Element Wood had their fortune told fifth.
:- tuple(I, wood), tuple(J, 5), I!=J.

%Earl's lucky element is Fire.
:- tuple(I, earl), tuple(J, fire), I!=J.

%Earl arrived immediately before the person with the Rooster.
:- tuple(I, earl), tuple(J, rooster), tuple(I, X), tuple(J, Y),
etype(A, rank), element(A, X), element(A, Y), X != Y-1.

%The person with the Dragon had their fortune told fourth.
:- tuple(I, dragon), tuple(J, 4), I!=J.
```

# Evaluation

- Clues: 800 clues were selected. Standard 10 fold cross validation was used.
  - Precision measures the number of correctly translated clues, save for permutations in the body of the rules, or head of disjunctive rules.
  - Recall measures the number of correct exact translations.

| Precision | Recall | F-measure |
|---|---|---|
| 87.64 | 86.12 | 86.87 |

# Evaluation

- Accuracy measures the number of correctly solved puzzles.
  - A puzzle is considered correctly solved if it provides a single correct solution.
  - 10-s, 15-s, 20-s gives results were the best possible set of 10, 15 and 20 puzzles was selected for training
  - Note 0.87 ^ 10 = 0.25

| | Accuracy |
|---|---|
| 10-fold cross-validation | 56% |
| Train with selected 10 | 22/40 (55%) |
| Train with selected 15 | 24/35 (68.75%) |
| Train with selected 20- | 25/30 (83.33%) |

# Recap



- More general fact extraction: From traditional fact extraction to a query based approach
  - Example of a traditional approach - YAPPIE
  - Generalizing text extraction – querying annotated parse trees
- Examples of decision making
  - How do processes materialize: building pathways
  - Should these drugs be prescribed at the same time? Can they interact?
- Obtaining more general knowledge from text needed for NLU.
- Conclusion

# Conclusion



- Need to push the envelop in NLP application to biological and clinical decision making
  - Treat generically processed text as semi-structured data and extraction as asking appropriate queries.
  - Obtain general knowledge from text
- We are at a stage where we can envision
  - Translating natural language text to a formal logic
  - And reason with that logic as a step towards natural language understanding

# Acknowledgements




- Collaborators: my students, post-doctoral researchers and colleagues
  - Especially, Prof. Graciela Gonzalez, Dr. Luis Tari, Dr. Joerg Hakenberg, Bob Leaman and Vo Nguyen.
- Funding agencies
  - NSF
  - Science Foundation of Arizona
  - ASU
  - IARPA
  - ONR