



语义解析在工业界的应用

刘占亮 | 一览群智



0.1

公司介绍

0.2

简介

0.3

语义解析

0.4

应用场景



Turning AI into productivity

一览群智

一览群智是一家以**自然语言处理**和**知识图谱**等认知智能技术为核心的人工智能公司，为客户提供一站式AI产品和行业解决方案。公司自主研发出智语、智图、智慧、智策四大产品，满足企业在超大规模多源异构情况下的数据治理融合、不同场景下的AI建模，和复杂决策分析需求，让AI技术快速在客户场景落地，打造客户专属的智能决策平台。



金融行业



公共安全



媒体情报





0.1

公司介绍

0.2

简介

0.3

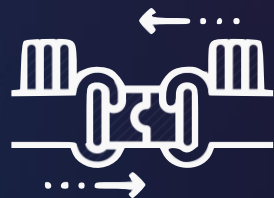
语义解析

0.4

应用场景

在构建智能搜索系统以及适用自然语言的用户交互接口时，语义解析技术的使用已经成为了一个很重要且很有效的模块。语义解析器可以将非统一化和非标准的人类自然语言与抽象严谨的逻辑表达式进行关联与映射，而逻辑表达式是一个将很多语法语义进行规范和完整地表达的经典方式。本报告将介绍语义解析领域的最新进展，以及其在一览群智AI行业落地中的应用。

▶ 搜索是什么



文档与查询的匹配



查询理解



文档理解



0.1

公司介绍

0.2

简介

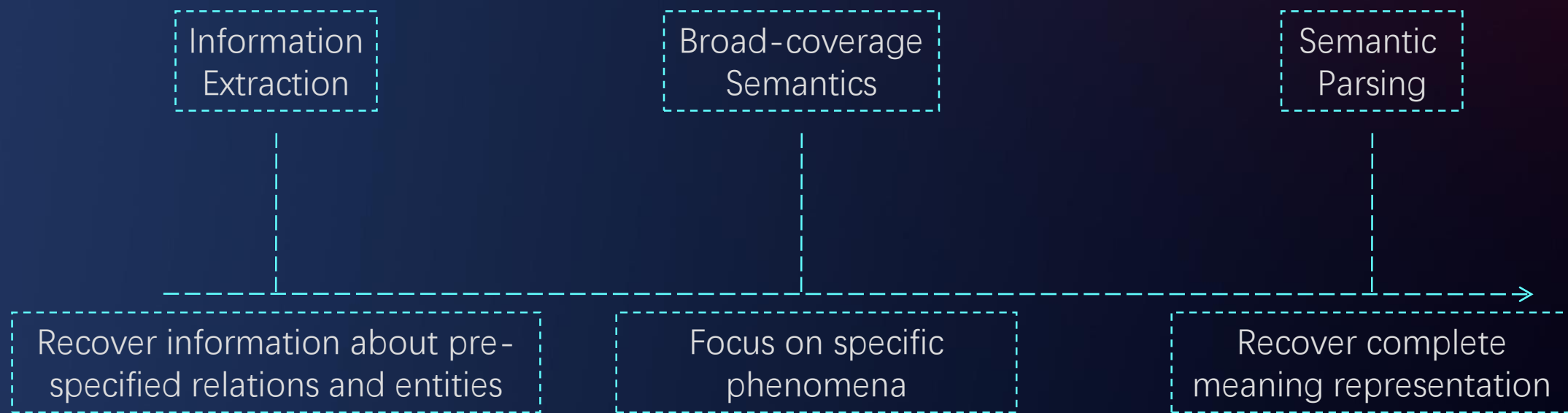
0.3

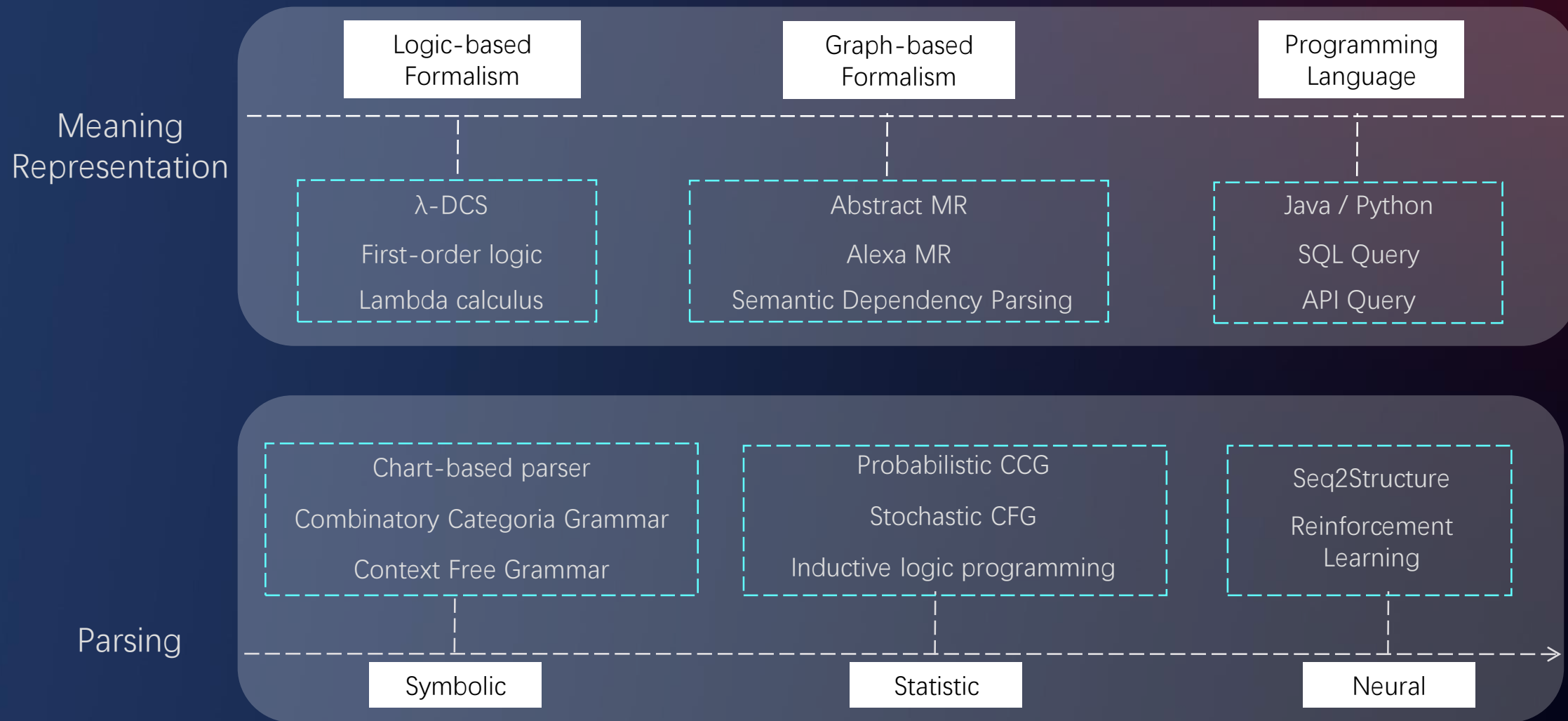
语义解析

0.4

应用场景

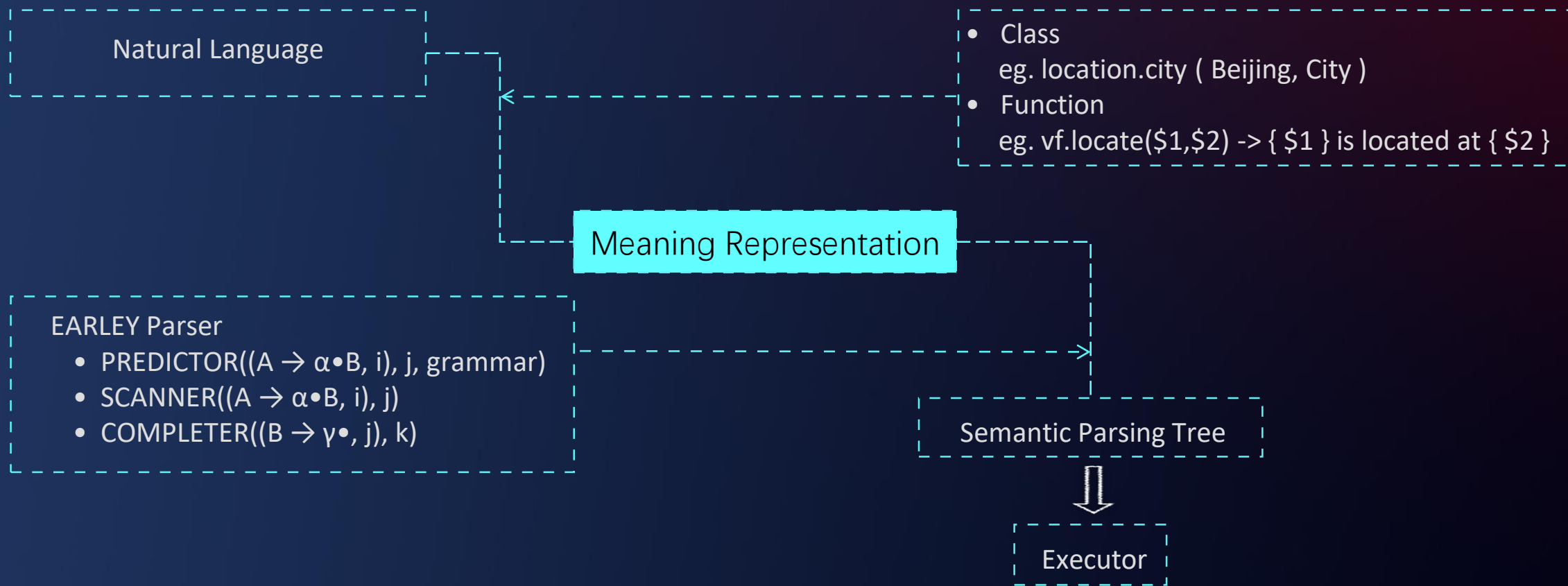
语义解析(Semantic Parsing): 把自然语言句子映射成逻辑形式的表达式。是问答、对话、自然语言理解等任务的核心技术。近年来, 随着深度学习技术和知识图谱技术的发展, 语义解析取得了长足的进展。







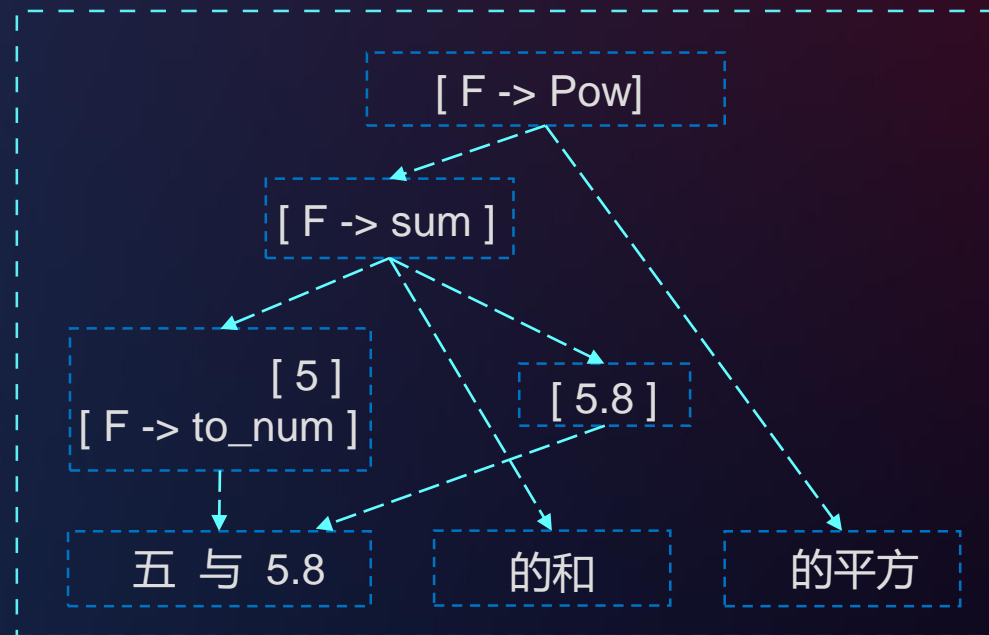
符号语义解析





数字解析语法示例

五与5.8的和平方的平方



```
nf.math.pow(  
  nf.math.sum(  
    5,  
    nf.math.to_number("5.8")  
  ),  
  2  
);
```



116.64

数字解析语法示例

中文数字语法

```
1 <cn_unit>="一"{nf.I(1)}|"二"{nf.I(2)}|"九"{nf.I(9)}|"壹"{nf.I(1)}|"貳"{nf.I(2)}...;
2 <cn_zero>="零"{nf.I(0)}|"〇"{nf.I(0)};
3 <cn_digit>=<cn_unit>{nf.I($1)}|<cn_zero>{nf.I($1)}|<digits>{nf.I($1)};
4 <numbers>=<cn_digit>{nf.I($1)}|<cn_digit> <numbers>{nf.util.concat($1, $2)};
5 <cn_e1>="十"{nf.I(10)}|"拾"{nf.I(10)}; <cn_e2>="百"{nf.I(100)}|"佰"{nf.I(100)};
6 <cn_e3>="千"{nf.I(1000)}|"仟"{nf.I(1000)}; <cn_e4>="万"{nf.I(10000)};
7 <cn_e8>="亿"{nf.I(100000000)}|"万万"{nf.I(100000000)};
8
9 <cn_e1s>=<cn_e1>{nf.I($1)}|<cn_e1> <cn_unit>{nf.math.sum($1,$2)}
10   |<cn_unit> <cn_e1>{nf.math.mul($1,$2)}
11   |<cn_unit> <cn_e1> <cn_unit>{nf.math.sum(nf.math.mul($1,$2),$3)}
12   |<cn_unit>{nf.I($1)}|<cn_zero>{nf.I($1)};
13 <cn_e2s>=<cn_unit> <cn_e2>{nf.math.mul($1,$2)}|<cn_e1s>{nf.I($1)}
14   |<cn_unit> <cn_e2> <cn_unit>{nf.math.sum(nf.math.mul($1,$2),nf.math.mul(10,$3))}
15   |<cn_unit> <cn_e2> <cn_e1s>{nf.math.sum(nf.math.mul($1,$2),$3)}
16   |<cn_unit> <cn_e2> <cn_zero> <cn_e1s>{nf.math.sum(nf.math.mul($1,$2),$4)};
17 <cn_e3s>=<cn_unit> <cn_e3>{nf.math.mul($1,$2)}|<cn_e2s>{nf.I($1)}
18   |<cn_unit> <cn_e3> <cn_unit>{nf.math.sum(nf.math.mul($1,$2),nf.math.mul(100,$3))}
19   |<cn_unit> <cn_e3> <cn_e2s>{nf.math.sum(nf.math.mul($1,$2),$3)}
20   |<cn_unit> <cn_e3> <cn_zero> <cn_e2s>{nf.math.sum(nf.math.mul($1,$2),$4)};
21 <cn_e4s>=<cn_e3s> <cn_e4>{nf.math.mul($1,$2)}|<cn_e3s>{nf.I($1)}
22   |<cn_e3s> <cn_e4> <cn_unit>{nf.math.sum(nf.math.mul($1,$2),nf.math.mul(1000,$3))}
23   |<cn_e3s> <cn_e4> <cn_e3s>{nf.math.sum(nf.math.mul($1,$2),$3)}
24   |<cn_e3s> <cn_e4> <cn_zero> <cn_e3s>{nf.math.sum(nf.math.mul($1,$2),$4)};
25 <cn_e8s>=<cn_e4s> <cn_e8>{nf.math.mul($1,$2)}|<cn_e4s>{nf.I($1)}
26   |<cn_e4s> <cn_e8> <cn_unit>{nf.math.sum(nf.math.mul($1,$2),nf.math.mul(10000000,$3))}
27   |<cn_e4s> <cn_e8> <cn_e4s>{nf.math.sum(nf.math.mul($1,$2),$3)}
28   |<cn_e4s> <cn_e8> <cn_zero> <cn_e4s>{nf.math.sum(nf.math.mul($1,$2),$4)};
29
30 <cn_decimal>="点" <numbers>{nf.math.decimal($2)};
31 <number>=<cn_e8s>{nf.I($1)}|<cn_decimal>{nf.I($1)}|<cn_e8s> <cn_decimal>{nf.math.sum($1,$2)};
32 <number>=<cn_e2>{nf.I($1)}|<cn_e3>{nf.I($1)}|<cn_e4> {nf.I($1)}|<cn_e8>{nf.I($1)};
```

英文数字语法

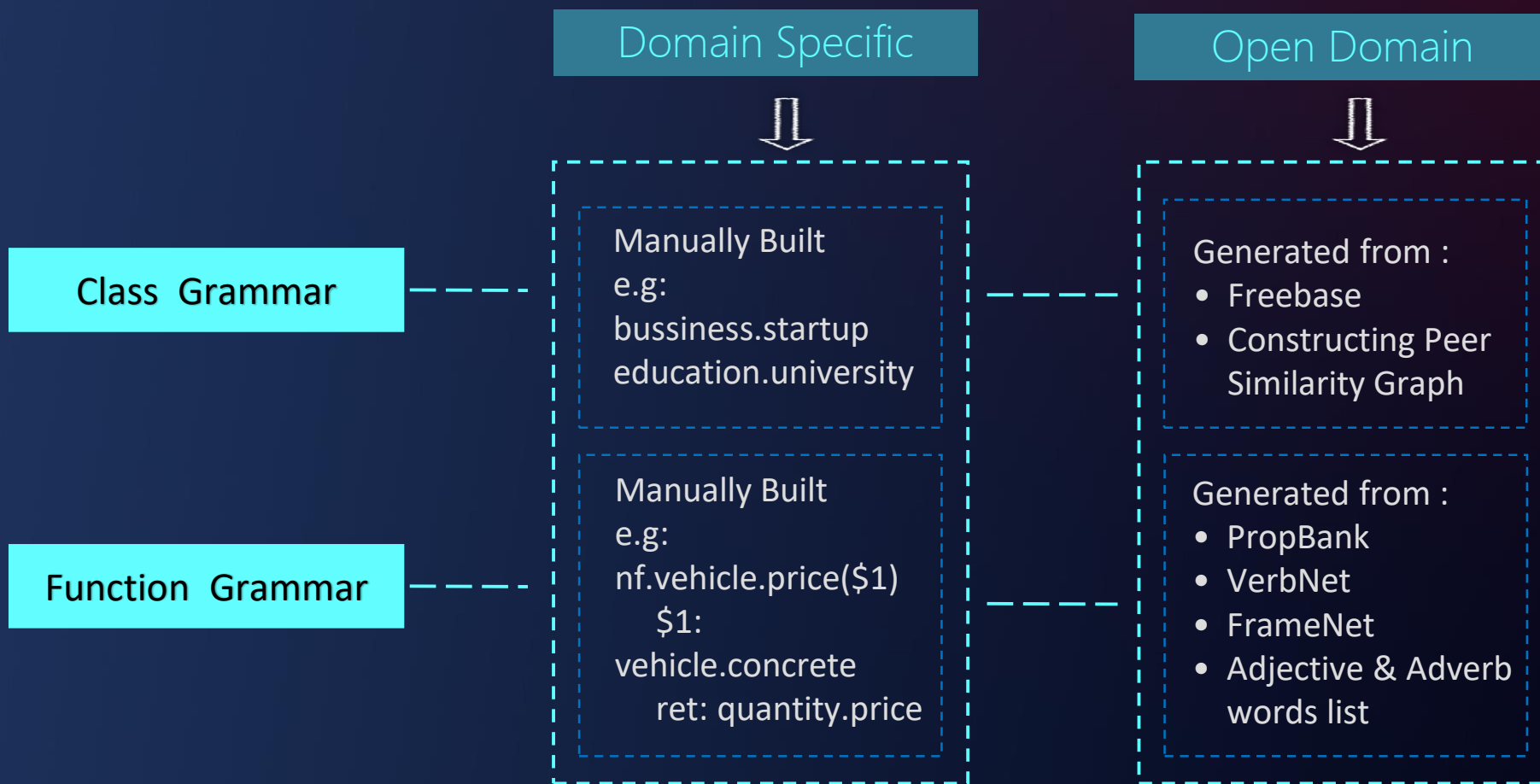
```
1 <unit>="zero"{nf.I(0)}|"one"{nf.I(1)}|"two"{nf.I(2)}|"three"{nf.I(3)} ...
2 <ten>="ten"{nf.I(10)}|"eleven"{nf.I(11)}|"twelve"{nf.I(12)} ... |"nineteen"{nf.I(19)};
3 <tens>="twenty"{nf.I(20)}|"thirty"{nf.I(30)}|"forty"{nf.I(40)} ... |"ninety"{nf.I(90)};
4
5 <e1>=<unit>{nf.I($1)}|<ten>{nf.I($1)}|<tens>{nf.I($1)}
6   |<tens> <unit>{nf.math.sum($1,$2)}
7   |<tens> "-" <unit>{nf.math.sum($1,$3)};
8 <e2>="hundred"{nf.I(100)}; <e3>="thousand"{nf.I(1000)};
9 <e6>="million"{nf.I(1000000)}; <e9>="billion"{nf.I(1000000000)};
10
11 <e2s>=<e1>{nf.I($1)}|<e2> {nf.I($1)}|<e1> <e2>{nf.math.mul($1,$2)}
12   |<e1> <e2> <e1> {nf.math.sum(nf.math.mul($1,$2),$3)}
13   |<e1> <e2> "and" <e1> {nf.math.sum(nf.math.mul($1,$2),$4)};
14
15 <e3s>=<e2s>{nf.I($1)}|<e3>{nf.I($1)}|<e2s> <e3>{nf.math.mul($1,$2)}
16   |<e2s> <e3> <e2s> {nf.math.sum(nf.math.mul($1,$2),$3)}
17   |<e2s> <e3> "and" <e2s> {nf.math.sum(nf.math.mul($1,$2),$4)};
18
19 <e6s>=<e3s>{nf.I($1)}|<e6>{nf.I($1)}|<e3s> <e6>{nf.math.mul($1,$2)}
20   |<e3s> <e6> <e3s> {nf.math.sum(nf.math.mul($1,$2),$3)}
21   |<e3s> <e6> "and" <e3s> {nf.math.sum(nf.math.mul($1,$2),$4)};
22
23 <e9s>=<e6s>{nf.I($1)}|<e9>{nf.I($1)}|<e6s> <e9>{nf.math.mul($1,$2)}
24   |<e6s> <e9> <e6s> {nf.math.sum(nf.math.mul($1,$2),$3)}
25   |<e6s> <e9> "and" <e6s> {nf.math.sum(nf.math.mul($1,$2),$4)};
26
27 <en_decimal> = "point" <numbers> {nf.math.decimal($2)};
28 <en_number> = <e9s> {nf.I($1)};
29
30 <number> = <en_number> {nf.I($1)} | <en_decimal> {nf.I($1)}
31   | <en_number> <en_decimal> {nf.math.sum($1,$2)}
32   | <en_number> "and" <en_decimal> {nf.math.sum($1,$3)};
```



数字解析语法示例

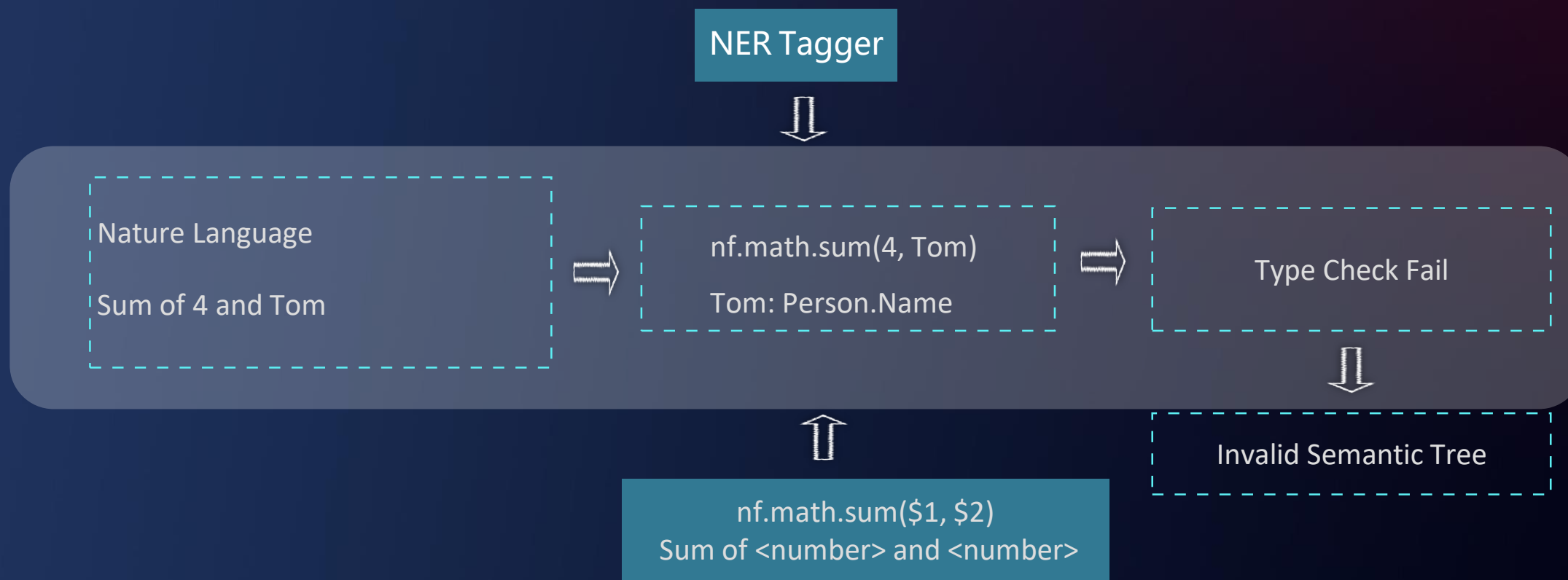
数字语义函数

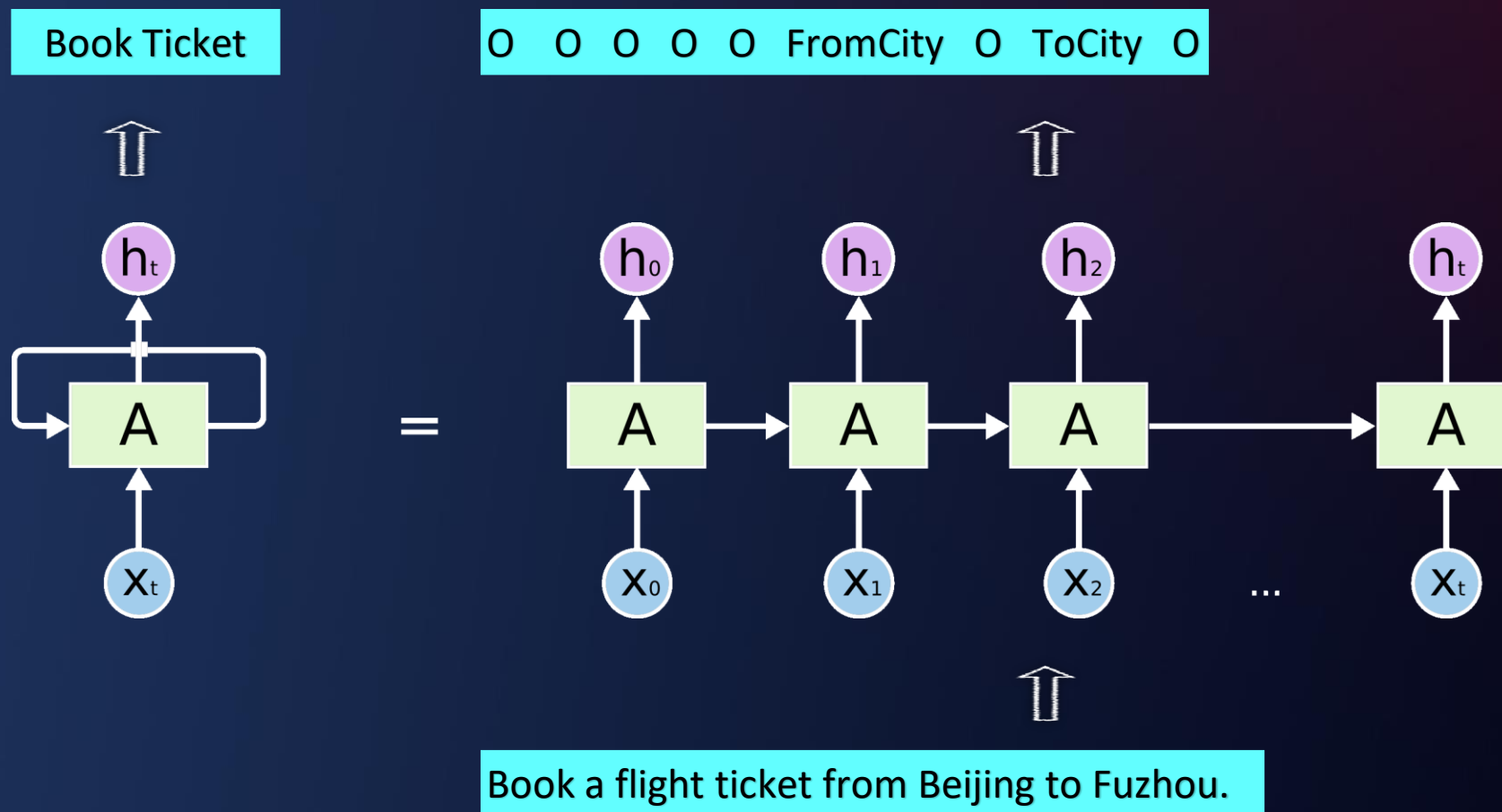
```
1 var nf = nf || {};  
2 nf.math = {};  
3  
4 nf.math.to_number = function(s) { return Number(s); }  
5  
6 nf.math.decimal = function(s) {  
7     s = s.toString();  
8     var n = Number(s);  
9     return n / Math.pow(10, s.length);  
10 }  
11  
12 nf.math.sum = function(x, y) { return x + y; }  
13 nf.math.sub = function(x, y) { return x - y; }  
14 nf.math.mul = function(x, y) { return x * y; }  
15 nf.math.div = function(x, y) { return x / y; }  
16 nf.math.neg = function(x) { return -x; }  
17 nf.math.pow = function(x, y) { return Math.pow(x, y); }
```



如何减少歧义

- Statistic Method: are often based on probabilistic CFGs (PCFGs) or probabilistic lexicalized CFGs trained on hand-labeled TreeBanks.
- Type-compatibility: The type of each child of a function node should match the corresponding argument type of the function.





Early stage

LUNAR

SHRDLU

Academic Effort



Cornell SPF



SEMPRE

Industrial Effort



OLAMI



Duckling



FMR by Golang



LUIS



IBM Watson



ALEXA



0.1

公司介绍

0.2

简介

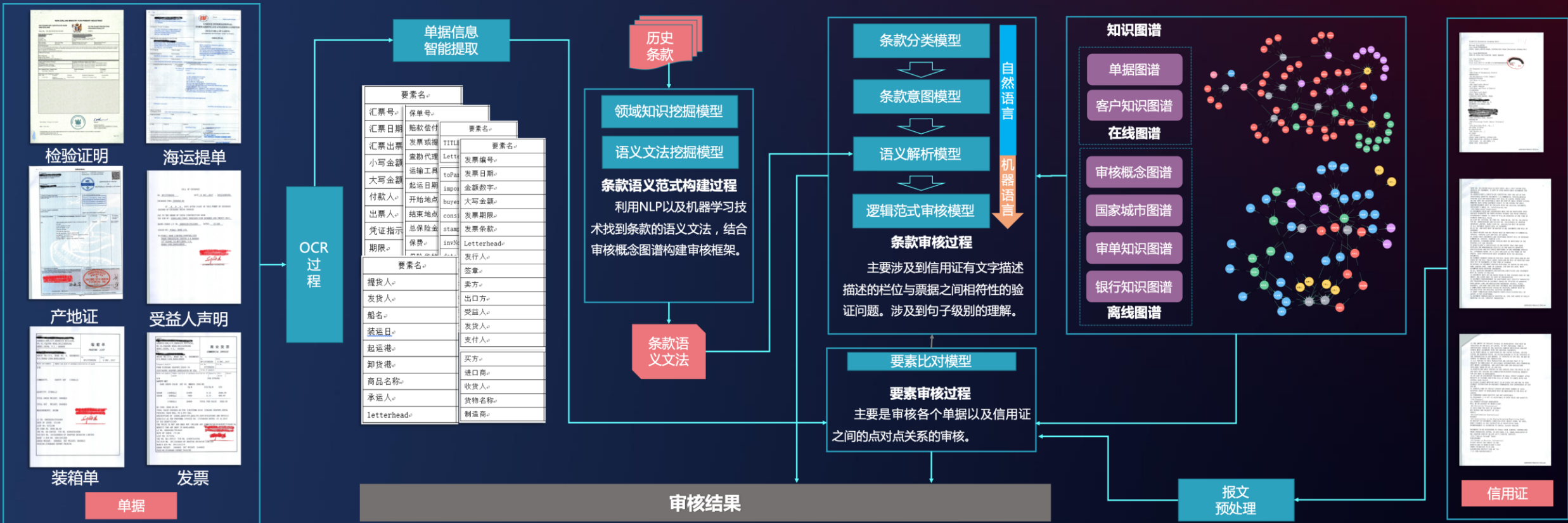
0.3

语义解析

0.4

应用场景

▶ 国际结算智能审单



输入条款

受益人以我方为付款人按100%发票金额出具即期汇票

意图识别

主要使用条款分类模型判断这句话涉及到的票据是哪一种

语义解析

受益人以我方为付款人按100%发票金额出具即期汇票

汇票对应的范式: BillOfExchange_Slots 【付款人, 金额, 汇票类型】

触发审核
函数

1. 付款人: 核查汇票当中的付款人必须是开证行。
2. 金额: 核查汇票的金额与信用证的金额是否相等。
3. 汇票类型: 核查汇票是否为即期汇票

受益人出具付款期限为提单日90日起的汇票

```
1 import pyfmr
2
3 def extractx():
4     p = pyfmr.Parser("./audit.grammar")
5     if p.grammar_index < 0:
6         print(p.error_message)
7         return
8
9     strx = [
10         "受益人以我方为付款人按100%发票金额出具即期汇票",
11         "即期汇票以伦敦花旗银行为付款人, 金额为发票金额",
12         "受益人开具以申请人为付款人的即期汇票",
13         "受益人出具付款期限为提单日90日起的汇票",
14     ]
15
16     for l in strx:
17         print(l)
18         for key in ["amount", "payer", "type", "duetime"]:
19             ret = p.extractx(l, key)
20             print(" %s:%s" % (key, ret))
21         print("="*60)
22
23
24 if __name__ == "__main__":
25     extractx()
```



```
1 #include "number.grammar"
2
3 <type> = "即期汇票" {nf.I("Bill of Exchange/at sight")}
4         | "汇票"      {nf.I("Bill of Exchange")}
5         ;
6
7 <amount> = "按" (any) "金额" {fmr.list(nf.I($2), "与发票金额一致性核验逻辑")}
8           | "金额为" (any) "金额" {nf.I($2)}
9           ;
10
11 <payer> = "以" (any) "为付款人" {nf.I($2)}
12         ;
13
14 <xdate> = "提单日" {nf.I("date1")}
15         ;
16
17 <duetime> = "付款期限为" <xdate> <number> "日起" {vf.time_before($3, $2)}
18         ;
```

```
type:["Bill of Exchange"]
duetime:[vf.time_before(
    nf.math.to_number("90"),
    "date1")]
```

经侦部门打击经济犯罪

- 随着科技技术的不断发展，各类经济犯罪活动不断增多、手段不断翻新，尤其是非法集资、传销、洗钱、骗取出口退税、网络赌博、知识产权等性质案件，严重危害人民利益和社会经济稳定。

• 政策方面

- 部局提出“两年锻造全新警种”、“信息化建设、数据化实战”的指导要求。
- 以数据引领经侦工作，实施数据导侦战略。

• 面临的问题

- 数据融合不足，非结构化数据以及其他警种的数据未得到有效整合。
- 数据处理手段不住，对于非结构化数据的处理仅限于检索，无法从中提取情报要素。
- 对数据的研判分析手段不足，对类罪的分析与管理手段不足，无法做到类罪的预测预警。

01
非法集资

案

E租宝、580亿,涉及90万用户，中晋系400亿，1.2参与者；
2018年7月130家P2P公司持续暴雷。

02
传销

件

善心汇，1000多亿，近600万人参与
“五行币” 过100亿、1040工程

03
洗钱

频

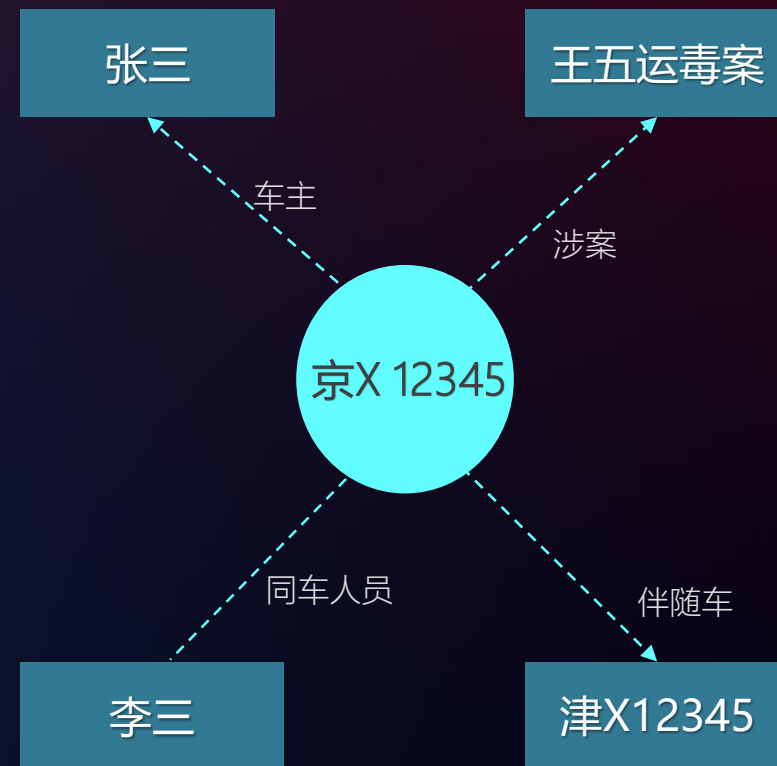
七台河9.8案件涉案金额过万亿，
上海特大操纵期货市场案，扰乱股票市场秩序

04
涉税

发

“鹰击1号”的跨地区特大虚开增值税发票案件，涉案金额近百亿；
厦门特大黄金票案涉案金额近百亿

▶ 案件智能搜索



▶ 地址标准化



▶ 地址标准化



► 语义解析-数字|日期|金额|时间

输入

壹万〇两拾一点一

五与5.8的和的1.5次方

解析

```
nf.math.sum(  
  nf.math.sum(  
    nf.math.mul(1, 10000),  
    nf.math.sum(  
      nf.math.mul(2, 10),  
      1  
    )  
  ),  
  nf.math.decimal(1)  
);
```

```
nf.math.pow(  
  nf.math.sum(  
    5,  
    nf.math.to_number("5.8")  
  ),  
  nf.math.to_number("1.5")  
);
```

执行

denotation: 10021.1

denotation: 35.49242172633476

<https://github.com/liuzl/fmr/blob/master/examples/math/grammars/number.en.grammar>

<https://github.com/liuzl/fmr/blob/master/examples/math/grammars/number.zh.grammar>

<https://mathsolver.zliu.org/>



让人工智能成为 生产力



欢迎加入！