The Interaction between Artificial Intelligence and Linguistics – a Historical Review and Prospect

刘群 LIU Qun

Huawei Noah's Ark Lab

Symposium on Humanities and Culture

2025.03.26-27

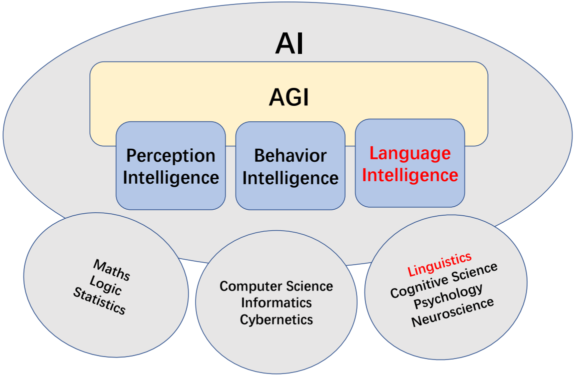
NOAH'S ARK LAB

HUAWEI

Content

Background

- Language is the advanced form of human intelligence, and language intelligence is an important part of artificial intelligence.
- Linguistics is considered to be one of the important theoretical foundations of artificial intelligence.
- In the history of artificial intelligence, linguistics has been deeply involved and played an important role.
- In the era of LLMs, it is necessary to re-examine the relation between linguistics and artificial intelligence.
- This talk is my attempt on this topic, as a long-term practitioner of AI, especially NLP, for decades.



Content

Timeline

- 1950 Turing Test
- 1954 First machine translation experiment
- 1957 Basic idea of distributional semantic (Firth)
- 1957 *Syntactic Structure* (Chomsky), transformational generative grammar
- 1959 *The Foundation of Structural Syntax* (Tesnière), dependent grammar
- 1962 Dartmouth Conference, Birth of Artificial Intelligence
- 1965 *Aspects of Syntactic Theory* (Chomsky)
- 1966 ALPAC Report, Funds in MT cut drastically
- 1967 Brown Corpus
- 1970 1970s-1980s Expert systems
- 1971 PoS tagging
- 1978 ARIAN78 Analysis-Transfer-Generation MT System
- 1984 CYC Encyclopedia Knowledge Base Project
- 1985 WordNet
- 1985 GPSG
- 1987 HPSG and LFG
- 1987 1st MUC, Information Extraction
- 1992 Penn Treebank
- 1993 Penn Discourse Treebank
- 1994 SCFG
- 1994 IBM SMT Models 1-5
- 1997 IBM Deep Blue Beated Kasparov
- 2000 FrameNet
- 2002 Semantic Role Labeling Task
- 2003 Phrase-based SMT
- 2005 PropBank
- 2006 Syntax-based SMT
- 2007 Dbpedia, Freebase
- 2011 IBM Watson beated Human in *Jeopardy*
- 2013 Word Embedding
- 2013 Seq2Seq Neural MT
- 2016 AlphaGo beated Lee Sedol
- 2017 Transformer Model
- 2018 Pre-trained Models: BERT, GPT etc.
- 2020 GPT-3 175B LLM
- 2022 ChatGPT

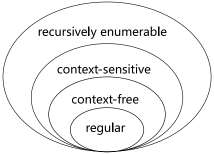
Content

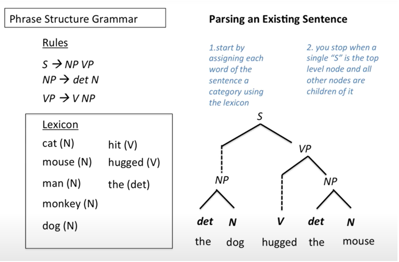
Timeline - The Early AI Stage

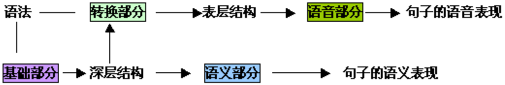
- 1950 Turing Test
- 1954 First machine translation experiment
- 1957 Basic idea of distributional semantic (Firth)
- 1957 *Syntactic Structure* (Chomsky), transformational generative grammar
- 1959 *The Foundation of Structural Syntax* (Tesnière), dependent grammar
- 1962 Dartmouth Conference, Birth of Artificial Intelligence
- 1965 *Aspects of Syntactic Theory* (Chomsky)
- 1966 ALPAC Report, Funds in MT cut drastically
- 1967 Brown Corpus
- 1970 1970s-1980s Expert systems
- 1971 PoS tagging
- 1978 ARIAN78 Analysis-Transfer-Generation MT System
- 1984 CYC Encyclopedia Knowledge Base Project
- 1985 WordNet
- 1985 GPSG
- 1987 HPSG and LFG
- 1987 1st MUC, Information Extraction
- 1992 Penn Treebank
- 1993 Penn Discourse Treebank
- 1994 SCFG
- 1994 IBM SMT Models 1-5
- 1997 IBM Deep Blue Beated Kasparov
- 2000 FrameNet
- 2002 Semantic Role Labeling Task
- 2003 Phrase-based SMT
- 2005 PropBank
- 2006 Syntax-based SMT
- 2007 Dbpedia, Freebase
- 2011 IBM Watson beated Human in *Jeopardy*
- 2013 Word Embedding
- 2013 Seq2Seq Neural MT
- 2016 AlphaGo beated Lee Sedol
- 2017 Transformer Model
- 2018 Pre-trained Models: BERT, GPT etc.
- 2020 GPT-3 175B LLM
- 2022 ChatGPT

Noam Chomsky’s Linguistic Theory

- Chomskys Hierarchy of Formal Languages
- Generative Grammar
- Aspects Model, Standard Theory
  - Deep Structure and Surface Structure
- Government and Binding Theory
  - $\bar{X}$ Theory
  - θ Theory
  - Case Theory
  - Binding Theory
  - Bounding Theory
  - Control Theory
  - Government Theory
- Minimalist Program



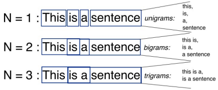


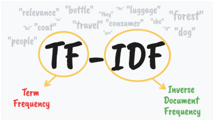


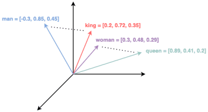
Distributed Semantics and its Influence

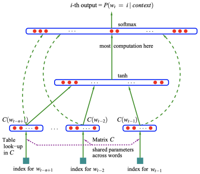
Firth, 1957: *You shall know a word by the company it keeps.* 你可以通过其伴随词来了解一个词的意思。

Bag of Words 词袋	N-gram LM N元语言模型	TF-IDF TF-IDF	Word Embedding 词嵌入	Neural LM 神经语言模型









HUAWEI NOAH'S ARK LAB

Content

Influence of Linguistics to Artificial Intelligence

- The Early AI Stage
- The Symbolic AI Stage
- The Statistical AI Stage
- The Neural AI Stage

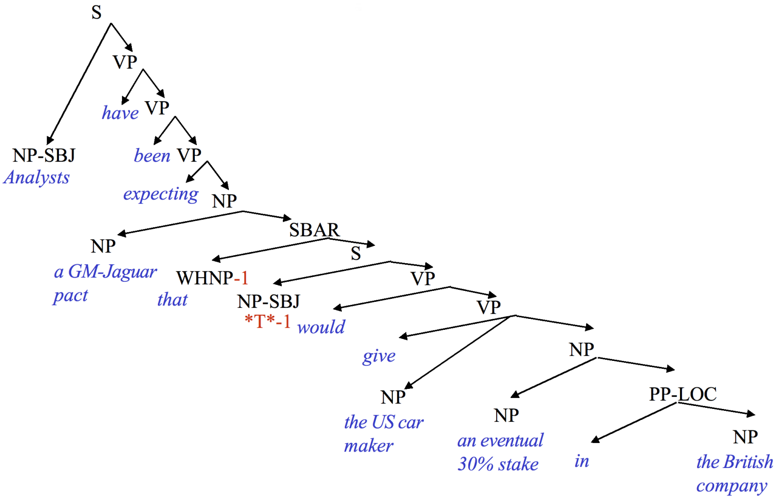
Timeline - The Symbolic AI Stage

- 1950 Turing Test
- 1954 First machine translation experiment
- 1957 Basic idea of distributional semantic (Firth)
- 1957 *Syntactic Structure* (Chomsky), transformational generative grammar
- 1959 *The Foundation of Structural Syntax* (Tesnière), dependent grammar
- 1962 Dartmouth Conference, Birth of Artificial Intelligence
- 1965 *Aspects of Syntactic Theory* (Chomsky)
- 1966 ALPAC Report, Funds in MT cut drastically
- 1967 Brown Corpus
- 1970 1970s-1980s Expert systems
- 1971 PoS tagging
- 1978 ARIAN78 Analysis-Transfer-Generation MT System
- 1984 CYC Encyclopedia Knowledge Base Project
- 1985 WordNet
- 1985 GPSG
- 1987 HPSG and LFG
- 1987 1st MUC, Information Extraction
- 1992 Penn Treebank
- 1993 Penn Discourse Treebank
- 1994 SCFG
- 1994 IBM SMT Models 1-5
- 1997 IBM Deep Blue Beated Kasparov
- 2000 FrameNet
- 2002 Semantic Role Labeling Task
- 2003 Phrase-based SMT
- 2005 PropBank
- 2006 Syntax-based SMT
- 2007 Dbpedia, Freebase
- 2011 IBM Watson beated Human in *Jeopardy*
- 2013 Word Embedding
- 2013 Seq2Seq Neural MT
- 2016 AlphaGo beated Lee Sedol
- 2017 Transformer Model
- 2018 Pre-trained Models: BERT, GPT etc.
- 2020 GPT-3 175B LLM
- 2022 ChatGPT

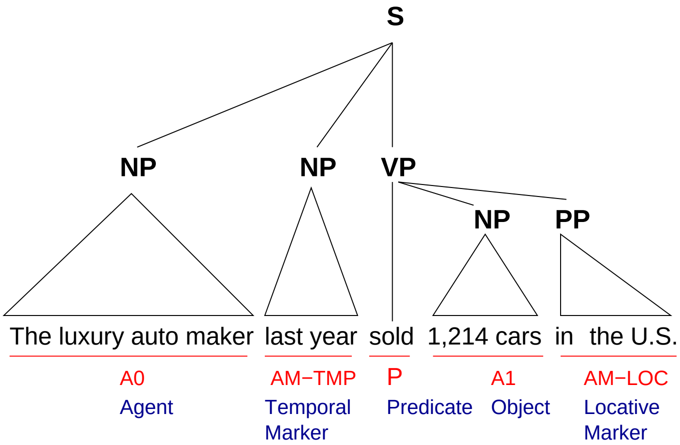
Penn Tree Bank and Its Derived Corpus

1992	Penn Treebank	First published in 1992, containing about 1 million words from the Wall Street Journal text , marked with syntactic structure.
2002	RST Discourse Treebank	Rhetoric Structure Theory (RST) Discourse Tree Bank, containing 385 articles from Penn Treebank and annotating discourse structures in the RST framework, as well as artificially generated excerpts and abstracts associated with source documents.
2002	Penn Chinese Treebank	In 2002, a Large-Scale Annotated Chinese Corpus was released to analyze Chinese text based on the syntax annotation method of Penn Treebank.
2004	NomBank	Released in 2004, providing semantic role annotations for noun phrases.
2005	PropBank	Released in 2005, providing semantic role annotations for English verbs.
2006	TimeBank	Released in 2006, providing detailed semantic annotations for time expressions.
2008	Penn Discourse Treebank (PDTB) 2.0	Released in 2008, containing a corpus of dialogue text, providing syntactic and semantic structural annotations at the discourse level.
2015	Universal Dependencies	Released version 1.0 in 2015, a multi-lingual syntactic annotation project, partly based on Penn Treebank.

Penn Tree Bank

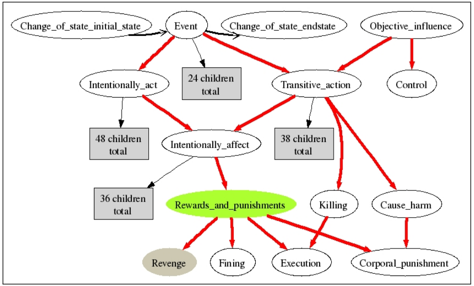


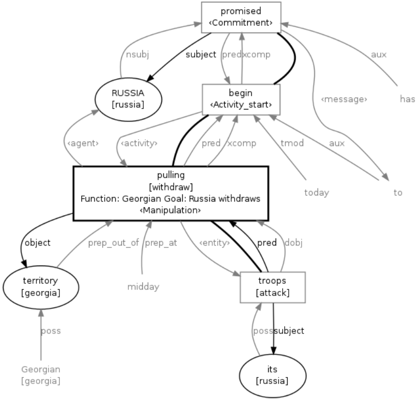
PropBank



HUAWEI NOAH'S ARK LAB

FrameNet

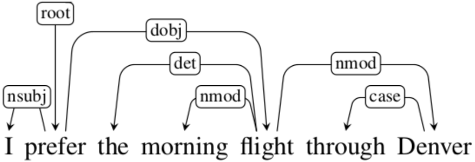




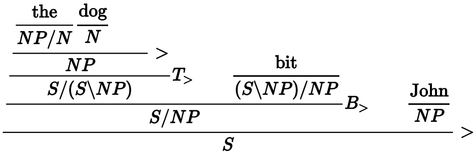
RUSSIA has promised to begin pulling its troops out of Georgia at midday today.

Dependency Grammar、Valence Grammar, Combinatorial Category Grammar（CCG）

- Dependency grammar is the simplest form of grammar: it only needs to establish dependencies between words, without marking words or phrases linguistically.
- Valence grammar introduces the concept of "valence" borrowed from chemistry into linguistics, to describe the semantics of words. The valence description of words can be a good supplement to the dependency grammar.
- Combinational category grammar gives each word a complex category representation, while the combination of words is as simple as an eliminating rule.
- All the above three grammars belong to lexicalized grammars, with which, we describe languages mainly using lexicons, rather than constructing complex combination rules like in phrase structure grammar.
- Because of their simple forms, these grammars have been studied and applied in NLP. In particular, dependency grammar is one of the most widely used language analysis tools.



Samples of Dependency Trees

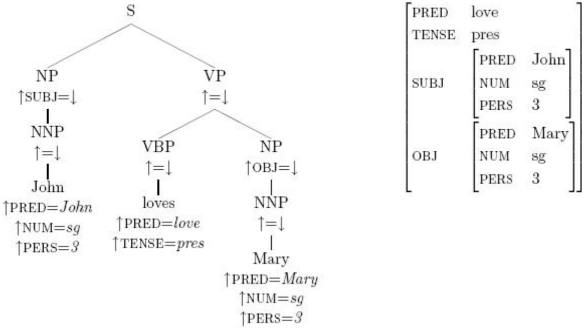


Samples of CCG Parsing

Unification-based Grammars

- From 1980s to 1990s, a number of new grammar theories have been put forward in computational linguistics, including lexical functional grammar (LFG), functional unification grammar (FUG), generalized phrase structure grammar (GPSG), head-driven phrase structure grammar (HPSG), etc.
- A common feature of these grammars is that they all use the form of complex feature sets + unification operations, so they are also called "unification-based grammars".
- Similar to dependency grammars, unification-based grammars do not use complex composition rules, but only use lexicons to describe the use of words. The complex feature sets can describe the linguistic features of words in details, and the unification operation has the advantages of order independence and monotony. This kind of grammar once received a lot of attentions and had a great influence.

FIGURE 1: C-structure annotated with f-structure equations and the resulting f-

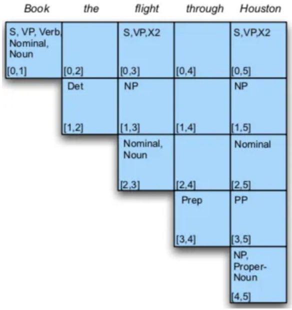


structure for the sentence *John Loves Mary*.

Samples of LFG Parsing

Syntatic Parsing Algorithms

	CFG	Dependency
Deterministic (for Compilers)	Recursive Descend LL（Top-down） Shift-Reduce LR（Bottom-up）	
Non-deterministic、Without probabilities	Recursive Descend、Shift-Reduce、 Chart、CYK、Tomita	
Probabilistic	Viterbi（PCFG Inference） Inside-Outside（PCFG Training）	Transition（Yamada、Nivre） Graph-based（MST）



CYK Parsing

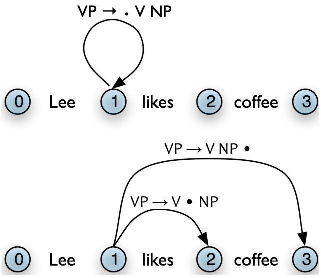
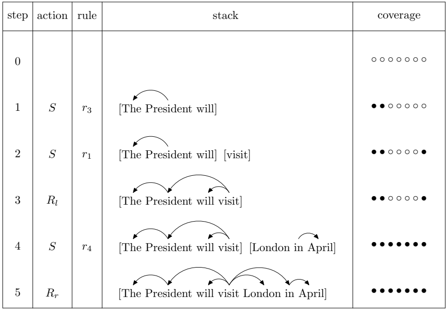


Chart Parsing



Shirt-Reduce Dependency Parsing

Content

Influence of Linguistics to Artificial Intelligence

- The Early AI Stage
- The Symbolic AI Stage
- The Statistical AI Stage
- The Neural AI Stage

Timeline - The Statistical AI Stage

- 1950 Turing Test
- 1954 First machine translation experiment
- 1957 Basic idea of distributional semantic (Firth)
- 1957 *Syntactic Structure* (Chomsky), transformational generative grammar
- 1959 *The Foundation of Structural Syntax* (Tesnière), dependent grammar
- 1962 Dartmouth Conference, Birth of Artificial Intelligence
- 1965 *Aspects of Syntactic Theory* (Chomsky)
- 1966 ALPAC Report, Funds in MT cut drastically
- 1967 Brown Corpus
- 1970 1970s-1980s Expert systems
- 1971 PoS tagging
- 1978 ARIAN78 Analysis-Transfer-Generation MT System
- 1984 CYC Encyclopedia Knowledge Base Project
- 1985 WordNet
- 1985 GPSG
- 1987 HPSG and LFG
- 1987 1st MUC, Information Extraction
- 1992 Penn Treebank
- 1993 Penn Discourse Treebank
- 1994 SCFG
- 1994 IBM SMT Models 1-5
- 1997 IBM Deep Blue Beated Kasparov
- 2000 FrameNet
- 2002 Semantic Role Labeling Task
- 2003 Phrase-based SMT
- 2005 PropBank
- 2006 Syntax-based SMT
- 2007 Dbpedia, Freebase
- 2011 IBM Watson beated Human in *Jeopardy*
- 2013 Word Embedding
- 2013 Seq2Seq Neural MT
- 2016 AlphaGo beated Lee Sedol
- 2017 Transformer Model
- 2018 Pre-trained Models: BERT, GPT etc.
- 2020 GPT-3 175B LLM
- 2022 ChatGPT

The Rise of Statistical Methods

- Linguistic-based methods (usually called rule-based methods) encountered bottlenecks in system performance when facing real language data in complex environments and are difficult to improve.
- In the early 1990s, IBM began to borrow statistical technologies from speech recognition to machine translation and carried out statistical machine translation research, which opened a new era of statistical NLP:
  - At the time, Fred Jelinek, head of machine translation at IBM, famously said: "Every time I fire a linguist, the performance of the speech recognizer goes up" (1998).
  - This statement has a great impact, and of course is very controversial. Fred Jelinek himself later gave some background explanations at a presentation in 2004.
- The statistical methods brought rapid performance improvement to NLP, but it also encountered bottlenecks quickly.
- Once again, there is a desire to introduce linguistics to improve the performance of the systems. So at this stage, more deep linguistic labeling corpus and more complex methods of combining statistics and linguistics have emerged.

NLP Methods Combining Statistics and Linguistics

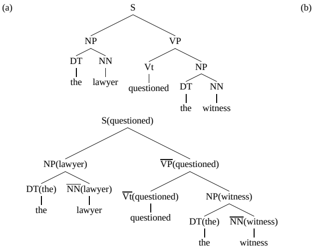


Figure 5: (a) A conventional parse tree as found for example in the Penn treebank. (b) A lexicalized parse tree for the same sentence. Note that each non-terminal in the tree now includes a single lexical item. For clarity we mark the head of each rule with an overline: for example for the rule NP → DT NN the child NN is the head, and hence the NN symbol is marked as $\overline{\text{NN}}$.

Lexicalized PCFG

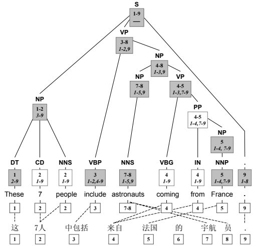


Figure 1: **Spans** and **complement-spans** determine what rules are extracted. Constituents in gray are members of the **frontier set**; a minimal rule is extracted from each of them.

String-to-Tree SMT

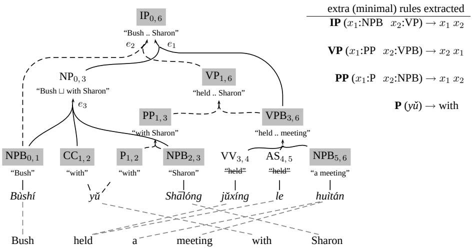


Figure 4: Forest-based rule extraction. Solid hyperedges correspond to the 1-best tree in Figure 3, while dashed hyperedges denote the alternative parse interpreting *yǔ* as a preposition in Figure 5.

Forest-to-String SMT

Content

Timeline - The Neural AI Stage

- 1950 Turing Test
- 1954 First machine translation experiment
- 1957 Basic idea of distributional semantic (Firth)
- 1957 *Syntactic Structure* (Chomsky), transformational generative grammar
- 1959 *The Foundation of Structural Syntax* (Tesnière), dependent grammar
- 1962 Dartmouth Conference, Birth of Artificial Intelligence
- 1965 *Aspects of Syntactic Theory* (Chomsky)
- 1966 ALPAC Report, Funds in MT cut drastically
- 1967 Brown Corpus
- 1970 1970s-1980s Expert systems
- 1971 PoS tagging
- 1978 ARIAN78 Analysis-Transfer-Generation MT System
- 1984 CYC Encyclopedia Knowledge Base Project
- 1985 WordNet
- 1985 GPSG
- 1987 HPSG and LFG
- 1987 1st MUC, Information Extraction
- 1992 Penn Treebank
- 1993 Penn Discourse Treebank
- 1994 SCFG
- 1994 IBM SMT Models 1-5
- 1997 IBM Deep Blue Beated Kasparov
- 2000 FrameNet
- 2002 Semantic Role Labeling Task
- 2003 Phrase-based SMT
- 2005 PropBank
- 2006 Syntax-based SMT
- 2007 Dbpedia, Freebase
- 2011 IBM Watson beated Human in *Jeopardy*
- 2013 Word Embedding
- 2013 Seq2Seq Neural MT
- 2016 AlphaGo beated Lee Sedol
- 2017 Transformer Model
- 2018 Pre-trained Models: BERT, GPT etc.
- 2020 GPT-3 175B LLM
- 2022 ChatGPT

Syntactic Ability of Neural LMs

- We propose to mask a word in BERT to observe the change of the hidden state of other words to predict the influence of one word on another. We find that the word influence matrix actually contains rich syntactic structure information.
- Recently, West Lake University and other institutions have found that only using the output layer hidden state of the LLMs, three simple methods can obtain the syntax analysis accuracy close to SotA, with very good cross-domain performance.

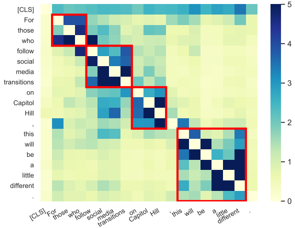


Figure 1: Heatmap of the impact matrix for the sentence "For those who follow social media transitions on Capitol Hill, this will be a little different."

	Model	LR	LP	F1
Non-LLM	SEPar♡	95.56	95.89	95.72
	SAPar♡	**96.19**	**96.61**	**96.40**
	TGCN♣	96.13	96.55	96.34
	LSTM★	-	-	88.30
	Transformer★	-	-	91.20
	GPT-2★	93.68	93.79	93.73
	OPT-6.7B★	94.63	94.52	94.58
LLM	LLaMA-7B★	95.50	95.12	95.31
	LLaMA-13B★	95.73	95.25	95.49
	LLaMA-33B★	96.05	95.56	95.81
	LLaMA-65B★	<u>96.09</u>	<u>95.72</u>	<u>95.90</u>
LLM[IT]	Alpaca-7B★	95.40	94.99	95.20
	Vicuna-7B★	95.37	94.93	95.16

Table 2: Fine-tuning results on PTB. LR: labeled recall. LP: labeled precision. ♡ means chart-based models. ♣ means transition-based models. ★ means sequence-based models. [IT] means instruction-tuned LLMs. The best results among all methods are **bolded** and the best sequence-based results are <u>underlined</u>.

Is linguistics useful for AI in the age of LLMs?

- Pre-trained LMs, especially LLMs, exhibit such strong NLU and NLG capabilities, so that we no longer resort to linguistic-based methods to improve NLP performance.
- Although LLMs no longer require direct linguistic knowledge in model design, we believe that linguistics can still play an important role in the era of LLMs:
  - Data engineering of the LLMs: The LLM pre-training data and instruction fine-tuning data play a decisive role in the capability of LLMs. However, the data engineering of LMs is still in the stage of experiential exploration and lacks clear theoretical guidance. Linguistics should play a role in this respect.
  - Evaluation of LLMs: The ability evaluation of LLMs is multi-dimensional, and the evaluation of language ability is also an important part of it. Linguistics should play a role in it.
  - Application of LLMs: The capability of LLMs depends more and more on the design of prompt words. Prompt word engineering has become an important means of LLM application, especially when agents based on LLMs are used to solve complex problems. Linguistics can help us a lot in this respect, which requires the comprehensive use of complex capabilities such as planning, memory, reflection, search and tool use of large language models.
  - Multi-agent application based on LLMs: Multi-agent has unique advantages in handling some complex problems. However, how multi-agents directly communicate and collaborate is an important constraint to the problem-solving capability of multi-agents. Linguistics should play an important role.

Common Sense Reasoning with LLMs Based on Situational Semantics

SMART: A Situation Model for Algebra Story Problems via Attributed Grammar

Yining Hong, Qing Li, Ran Gong, Daniel Ciao, Siyuan Huang, Song-Chun Zhu

University of California, Los Angeles, USA.

yininghong@cs.ucla.edu, {liqing, nikepupu, danielciao, huangsiyuan}@ucla.edu, sczhu@stat.ucla.edu

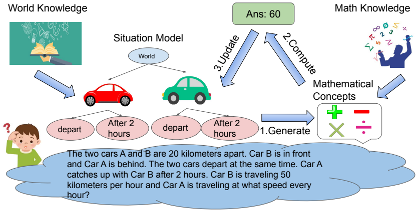


Figure 1: The process of human solving algebra story problems: We first hallucinate a situation model from the text and then perform arithmetic reasoning on the situation model to compute an answer. If we fail to generate a correct solution, we can adjust our situation model accordingly.

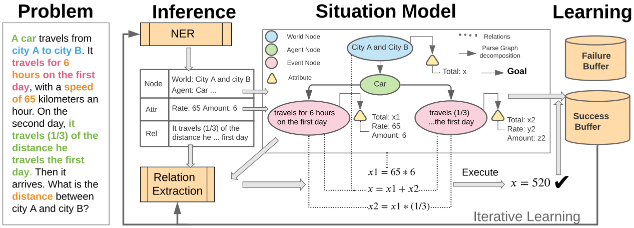


Figure 2: Overview of our SMART model. The Named Entity Recognition (NER) module extracts the spans of nodes, attributes, as well as relations from the text, and construct a parse graph using Attributed Grammar. The Relation Extraction module uses the relation spans and the parse graph already constructed to embed some relations into the parse graph. In the updated graph parser, Relation Extraction corresponds to Seq2Seq. The relations are then executed to get the final answer. If the answer is correct, it is added to the buffer of pseudo-gold parse graphs to train NER and Seq2Seq. If not, it is added to the failure set to be updated in the following iterations.

HUAWEI NOAH'S ARK LAB

KnowLogic: A Benchmark for Commonsense Reasoning via Knowledge-Driven Data Synthesis

KnowLogic: A Benchmark for Commonsense Reasoning via Knowledge-Driven Data Synthesis

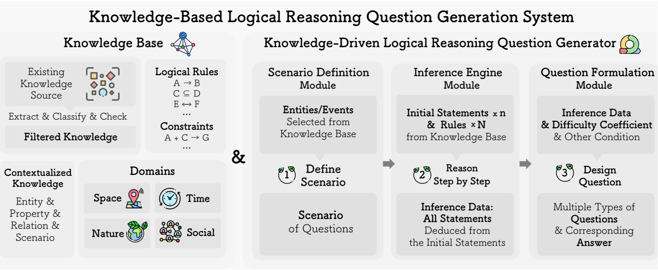
Weidong Zhan[1], Yue Wang[2], Nan Hu[1], Liming Xiao[1], Jingyuan Ma[2], Yuhang Qin[1], Zheng Li[2], Yixin Yang[2], Sirui Deng[1], Jinkun Ding[1], Wenhan Ma[2], Rui Li[2], Weilin Luo[3], Qun Liu[3], Zhifang Sui[2*]

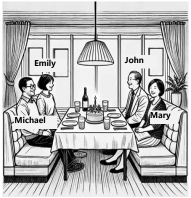
[1]Center for Chinese Linguistics, Department of Chinese Language and Literature, Peking University

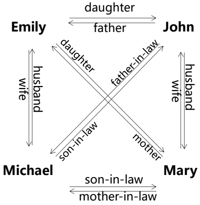
[2]School of Computer Science, State Key Laboratory of Multimedia Information Processing, Peking University

[3]Huawei Noah's Ark Lab, China

szf@pku.edu.cn, zwd@pku.edu.cn







Content

Background

Influence of Linguistics to Artificial Intelligence

Influence of Artificial Intelligence to Linguistics

Recent Debate between Chomsky and Hinton on ChatGPT

Conclusion

Quantitative Linguistics Research Based on Big Data

刘海涛教授计量语言学报告

发布者：xujiajin [发表时间]：2019-06-22 [来源]： [浏览次数]： 1106

2019年6月18日下午16:00-18:00，浙江大学刘海涛教授在110期语料库沙龙上做了题为“大数据时代语言研究的思考与践行”的学术报告。

刘教授以大数据为人类生活创造了前所未有的可量化维度为背景，提出大数据带给语言研究的机遇，计量语言学越来越受到学界青睐。同时，刘教授也向大家展示了基于语言大数据研究的挑战，即自然语言处理界对于语言学家作用及贡献的争议。随后，刘教授介绍了基于多语言依存距离进行的相关研究，展示了大数据时代计量语言学研究的最新成果，包括通过对依存距离的计算，揭示认知规律及语言普遍性、生态多样性与语言多样性之间的关系等。

报告后，刘教授与现场师生就大数据分析在商务文本中的应用、翻译文本质量测量、以及依存距离与短时记忆之间关系的研究等问题展开了讨论与互动。

24/6/27 00:14 Haitao Liu - Google 学术搜索

Haitao Liu

Professor of Linguistics, Zhejiang Univerity

Quantitative Linguistics
Digital Humanities
Dependency Grammar
Language Planning
Interlinguistics

	总计	2019 年至今
引用	4543	2564
h 指数	32	25
i10 指数	82	56

2 篇文章 5 篇文章

无法查看的文章 可查看的文章

根据资助方的强制性开放获取政策

标题	引用次数	年份
Dependency distance as a metric of language comprehension difficulty H Liu Journal of Cognitive Science 9 (2), 159-191	403	2008
Approaching human language with complex networks J Cong, H Liu Physics of life reviews 11 (4), 598-618	275	2014
Dependency distance: A new perspective on syntactic patterns in natural languages H Liu, C Xu, J Liang Physics of life reviews 21, 171-193	274	2017
Dependency direction as a means of word-order typology: A method based on dependency treebanks H Liu Lingua 120 (6), 1567-1578	173	2010
The effects of sentence length on dependency distance, dependency direction and the implications–based on a parallel English–Chinese dependency treebank J Jiang, H Liu Language Sciences 50, 93-104	164	2015

Dependency direction as a means of word-order typology: A method based on dependency treebanks

	VS	SV	VO	OV	NAdj	AdjN	WALS
Arabic (ara)	61.4 (2153)	38.6 (1351)	91 (5313)	9 (524)	95.9 (3953)	4.1 (167)	VS–VO–NAdj
Bulgarian (bul)	18.5 (3,036)	81.5 (13,417)	90.1 (6224)	9.9 (682)	1.6 (180)	98.4 (11,212)	?–VO–AdjN
Catalan (cat)	18.5 (4584)	81.5 (20,221)	85.5 (19,080)	14.5 (3239)	99.2 (1680)	0.8 (14)	?–VO–NAdj
Chinese (chi)	1.3 (19)	98.7 (1400)	98 (1679)	2 (34)	0.4 (2)	99.6 (461)	SV–VO–AdjN
Czech (cze)	27.4 (34,273)	72.6 (90,841)	72.9 (74,583)	27.1 (27,735)	8.6 (11,521)	91.4 (122,004)	SV–VO–AdjN
Danish (dan)	19.8 (1015)	80.2 (4122)	99.1 (8739)	0.9 (81)	60 (1683)	40 (1124)	SV–VO–AdjN
Dutch (dut)	28.7 (13,258)	71.3 (33,000)	82.5 (71,030)	17.5 (15,085)	7.4 (2024)	92.6 (25,207)	SV–?–AdjN
Greek (ell)	34.7 (1609)	65.3 (3029)	80.5 (3437)	19.5 (834)	8.4 (400)	91.6 (4345)	?–VO–AdjN
English (eng)	3.2 (1116)	96.8 (33,916)	93.5 (28,219)	6.5 (1959)	2.6 (661)	97.4 (24,801)	SV–VO–AdjN
Basque (eus)	20.4 (765)	79.6 (2990)	12.8 (381)	87.2 (2589)	78 (1234)	22 (349)	SV–OV–NAdj
German (ger)	33.2 (17,382)	66.8 (34,938)	36.8 (9447)	63.2 (16,237)	37.1 (15,355)	62.9 (26,016)	SV–?–AdjN
Hungarian (hun)	26.6 (1764)	73.4 (4862)	47.8 (2600)	52.2 (2843)	2.3 (339)	97.7 (14,239)	SV–?–AdjN
Italian (ita)	24.5 (869)	75.5 (2681)	82.3 (2090)	17.7 (451)	60.9 (2374)	39.1 (1523)	?–VO–NAdj
Japanese (jpn)	0	100 (5509)	0	100 (27,553)	0	100 (3820)	SV–OV–AdjN
Portuguese (por)	15.7 (1899)	84.3 (10,190)	85.1 (9447)	14.9 (1656)	70.1 (5858)	29.9 (2495)	SV–VO–NAdj
Romanian (rum)	21.9 (648)	78.1 (2313)	88.3 (1568)	11.7 (208)	66.9 (2905)	33.1 (1439)	SV–VO–NAdj
Slovenian (slv)	38.9 (658)	61.1 (1035)	74.5 (2375)	25.5 (815)	11 (189)	89 (1534)	SV–VO–AdjN
Spanish (spa)	21.5 (1107)	78.5 (4032)	77.3 (3417)	22.7 (1006)	98 (431)	2 (9)	?–VO–NAdj
Swedish (swe)	22.7 (4296)	77.3 (14,589)	94.6 (10,411)	5.4 (596)	0.4 (26)	99.6 (6656)	SV–VO–AdjN
Turkish (tur)	8.1 (284)	91.9 (3208)	4 (255)	96 (6175)	0.3 (11)	99.7 (3514)	SV–OV–AdjN

[a] In fact, here we use *dominant word order* unlike the definition in Croft (2002:60), and closer to the understanding of basic word order in Whaley (1997:100). In other words, it only shows that one of the word order types is more frequent (or dominant) in language use. Dryer (2008a) points out tha WALS also uses the *dominant word order* in this meaning, to emphasize that priority is given to the criterion of what is more frequent in language use

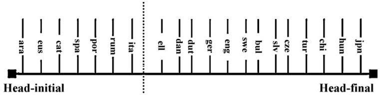


Fig. 5. 20 languages in Tesnière's typological classification system.

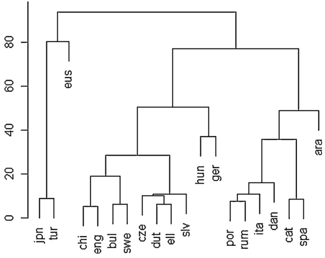


Fig. 10. Clustering of observations for 20 languages.

Deciphering Ancient Characters Based on SMT Model

A Computational Approach to Deciphering Unknown Scripts

Kevin Knight
USC/Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292
knight@isi.edu

Kenji Yamada
USC/Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292
kyamada@isi.edu

Figure 1: The Phaistos Disk (c. 1700BC). The disk is six inches wide, double-sided, and is the earliest known document printed with a form of movable type.

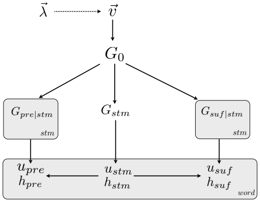
B → b or v
D → d
G → g
J → ñ
L → l l or y
a → a or á
b → b or v
d → d
e → e or é
f → f
g → g
i → i or í
l → l
m → m
n → n
o → o or ó
p → p

r → r
t → t
tS → c h
u → u or ú
x → j
nothing → h
T (followed by a, o, or u) → z
T (followed by e or i) → c or z
T (otherwise) → c
k (followed by e or i) → q u
k (followed by s) → x
k (otherwise) → c
rr (at beginning of word) → r
rr (otherwise) → rr
s (preceded by k) → nothing
s (otherwise) → s

A Statistical Model for Lost Language Decipherment

Benjamin Snyder and **Regina Barzilay**
CSAIL
Massachusetts Institute of Technology
{bsnyder,regina}@csail.mit.edu

Kevin Knight
ISI
University of Southern California
knight@isi.edu

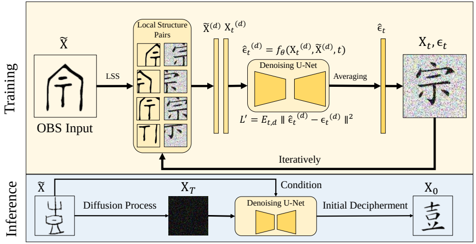


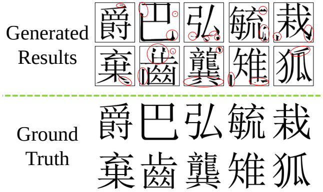
In this paper we propose a method for the automatic decipherment of lost languages. ...
We employ a non-parametric Bayesian framework to simultaneously capture both low-level character mappings and highlevel morphemic correspondences. ...
When applied to the ancient Semitic language Ugaritic, the model correctly maps 29 of 30 letters to their Hebrew counterparts, and deduces the correct Hebrew cognate for 60% of the Ugaritic words which have cognates in Hebrew.

Deciphering Oracle Bone Language with Diffusion Models

Deciphering Oracle Bone Language with Diffusion Models

Haisu Guan[1], **Huanxin Yang**[1], **Xinyu Wang**[2,*], **Shengwei Han**[3], **Yongge Liu**[3], **Lianwen Jin**[4], **Xiang Bai**[1], **Yuliang Liu**[1,*]
[1]Huazhong University of Science and Technology [2]The University of Adelaide
[3]Anyang Normal University [4]South China University of Technology
[1]{haisuguan, ylliu}@hust.edu.cn
*Corresponding authors





Language Generation and Evolution Research Based on Multi-Agents Interaction

Emergence and evolution of language in multi-agent systems

Dorota Lipowska [a,*], Adam Lipowski [b]

[a] *Faculty of Modern Languages and Literature, Adam Mickiewicz University, Poznań, Poland*
[b] *Faculty of Physics, Adam Mickiewicz University, Poznań, Poland*

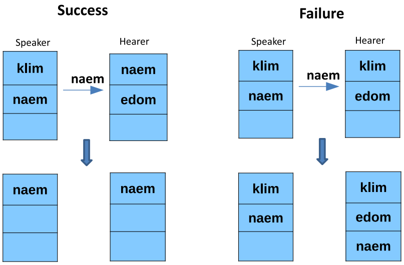


Fig. 1. An elementary step in the single-object version of the naming game.

Naming Game

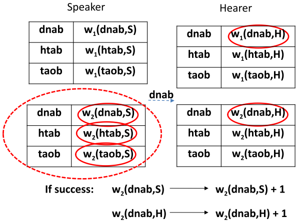


Fig. 2. An elementary step in a 2-object version of the signaling game model with reinforcement learning (Lipowska and Lipowski, 2018). The speaker randomly chooses an object (the corresponding section of the inventory is encircled by a dotted line). Using the relevant weights (in solid circles), the speaker selects one of its words (here: "dnab"). Next the hearer tries to guess the object the speaker is talking about, taking into account the weights of the communicated word (in circles). If the hearer's guess is correct, both agents increase their corresponding weights by 1. Otherwise, the weights remain unchanged.

Signaling Game with Reinforcement Learning

Content

Noam Chomsky's criticism to ChatGPT

The New York Times

GUEST ESSAY

Noam Chomsky: The False Promise of ChatGPT

March 8, 2023

By Noam Chomsky, Ian Roberts and Jeffrey Watumull

The human mind is not, like ChatGPT and its ilk, a lumbering statistical engine for pattern matching, gorging on hundreds of terabytes of data and extrapolating the most likely conversational response or most probable answer to a scientific question. On the contrary, the human mind is a surprisingly efficient and even elegant system that operates with small amounts of information; it seeks not to infer brute correlations among data points but to create explanations.

Indeed, such programs are stuck in a prehuman or nonhuman phase of cognitive evolution. Their deepest flaw is the absence of the most critical capacity of any intelligence: to say not only what is the case, what was the case and what will be the case — that's description and prediction — but also what is not the case and what could and could not be the case. Those are the ingredients of explanation, the mark of true intelligence.

Geoffery Hinton's criticism to Chomsky's Linguistics

Interview with Geoffrey Hinton from the 2024 Nobel Prize banquet

So there is a whole school of linguistics that comes from Chomsky that thinks that it's complete nonsense to say these things understand, that they don't process language at all in the same way as we do. I think that school is wrong. I think it's clear now that neural nets are much better at processing language than anything ever produced by the Chomsky School of Linguistics. But there's still a lot of debate about that, particularly among linguists.

[However, in another interview (https://www.youtube.com/watch?v=b_DUft-BdIE) when being asked "One of Chomsky's counter arguments to that the language models work the same as that we have sparse input for our understanding"] We're probably using some other learning algorithm. And in that sense, Chomsky may be right that we learn based on less knowledge.

Content

Interactions between AI and Linguistics in the Era of LLMs

- Some important linguistic concepts emerged alongside AI's birth
- Linguistic theories profoundly shaped AI, especially NLP
- Symbolic NLP systematically implements linguistic theories through structured resources (treebanks), analytical algorithms (dependency parsers), and applied systems (MT).
- Statistical methods both reduced linguistics' role and highlighted its value for solving complex problems
- LLMs exhibit dual impacts:
  - Traditional linguistics motivated methods marginalized
  - New opportunities: data engineering, evaluation, multi-agent systems
- The success of LLMs brings debate about LLMs and Linguistics
- AI empowers linguistics with new research tools

Thank you!

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home and organization for a fully connected, intelligent world.

HUAWEI