

Experiential Learning from Sequential Data



<https://agirussia.org>

Anton Kolonin

akolonin@aigents.com

Facebook: akolonin

Telegram: akolonin



SingularityNET



Motivation

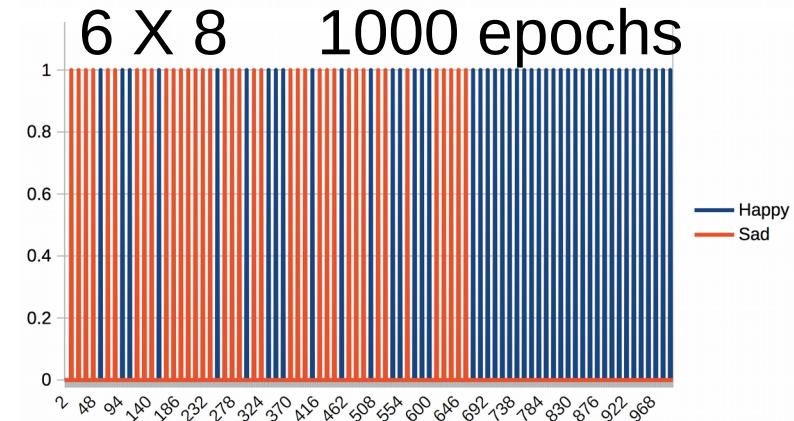
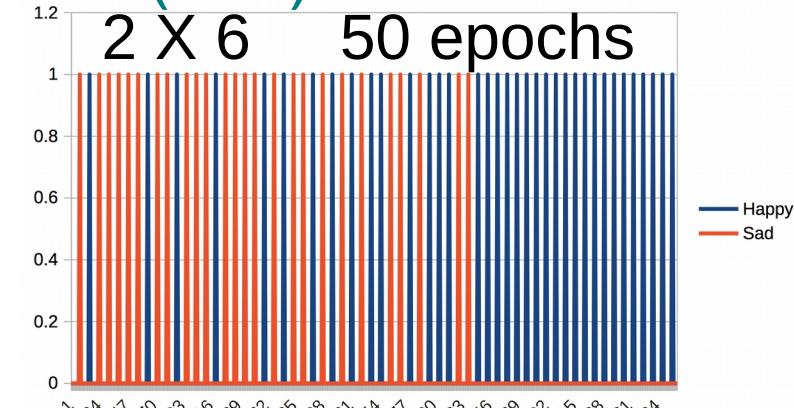
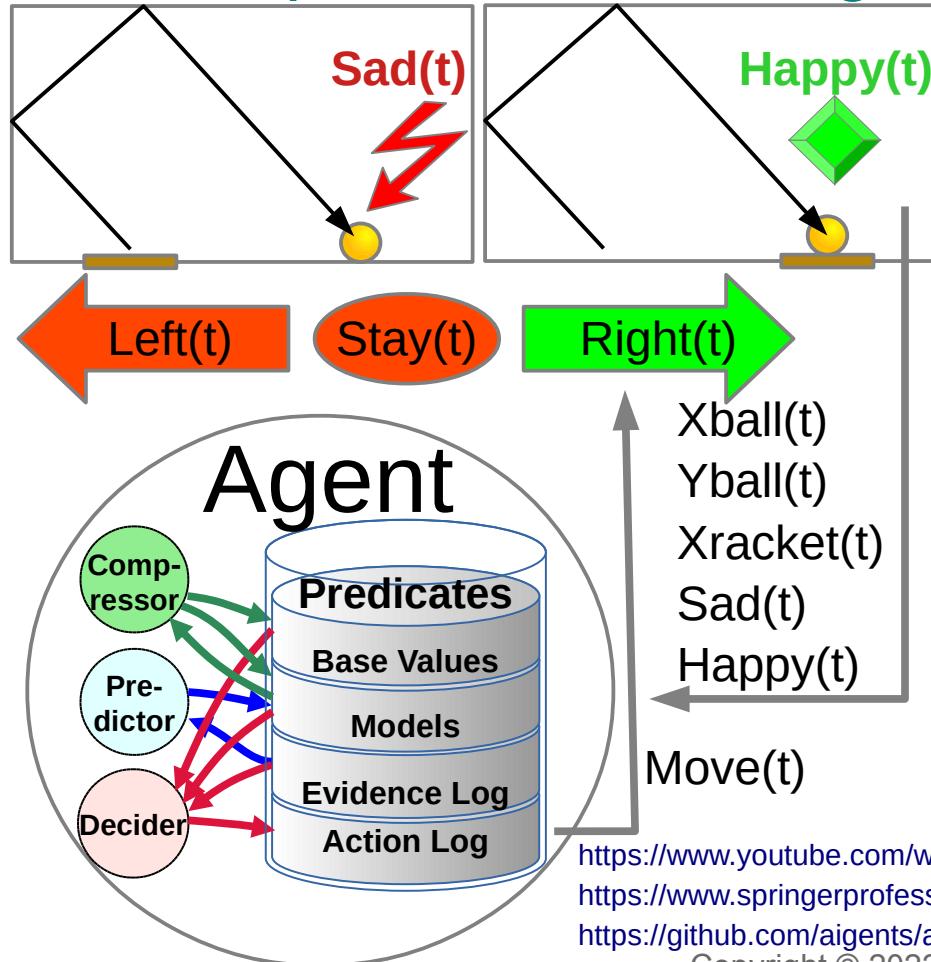
Identifying successful/unsuccessful sequential experiences for experiential learning for global (self)reinforcement

Discovering NLP patterns such as words and punctuation for further unsupervised language learning

<https://arxiv.org/abs/2205.11443>
<https://github.com/aigents/pygents>

Unsupervised Learning of Temporal Abstractions with Slot-based Transformers
Anand Gopalakrishnan, Kazuki Irie, Jürgen Schmidhuber, Sjoerd van Steenkiste
<https://arxiv.org/abs/2203.13573>

Identifying successful/unsuccessful sequential experiences for experiential learning with global (self)reinforcement

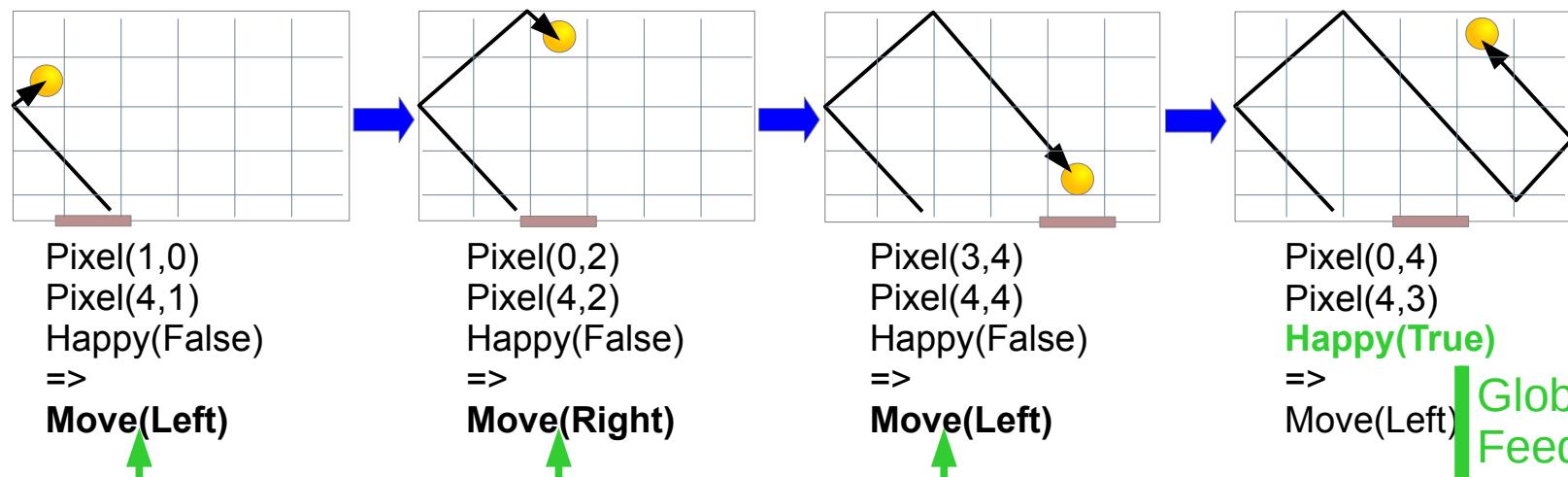
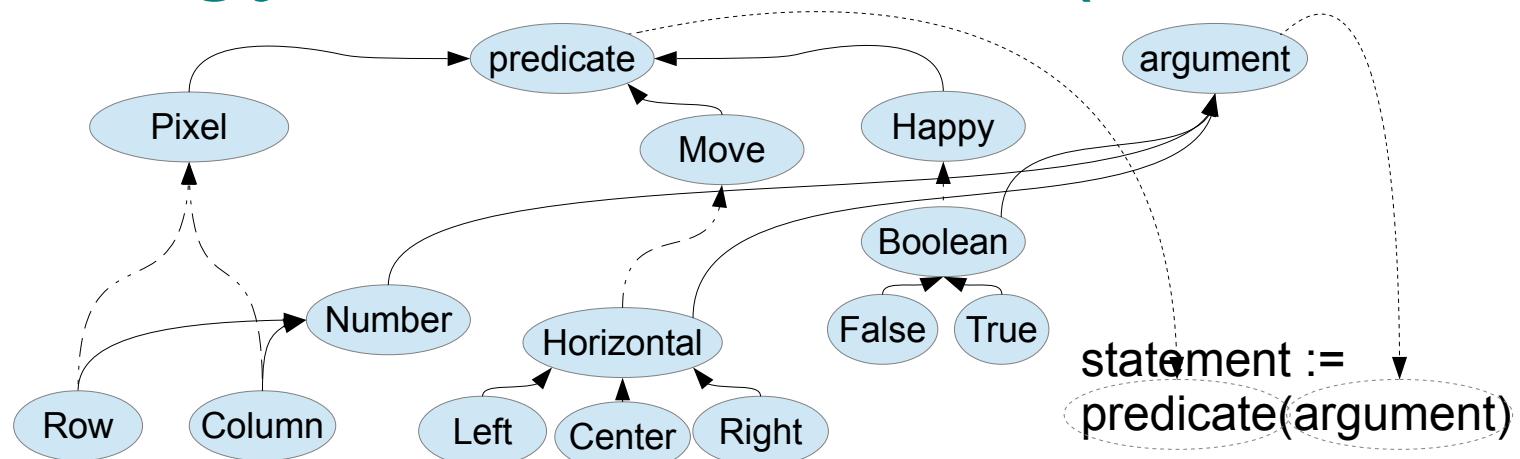


<https://www.youtube.com/watch?v=2LPLhJKh95g>

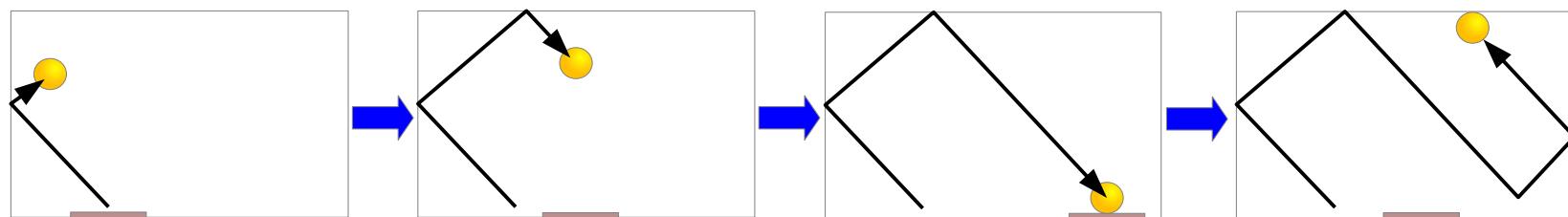
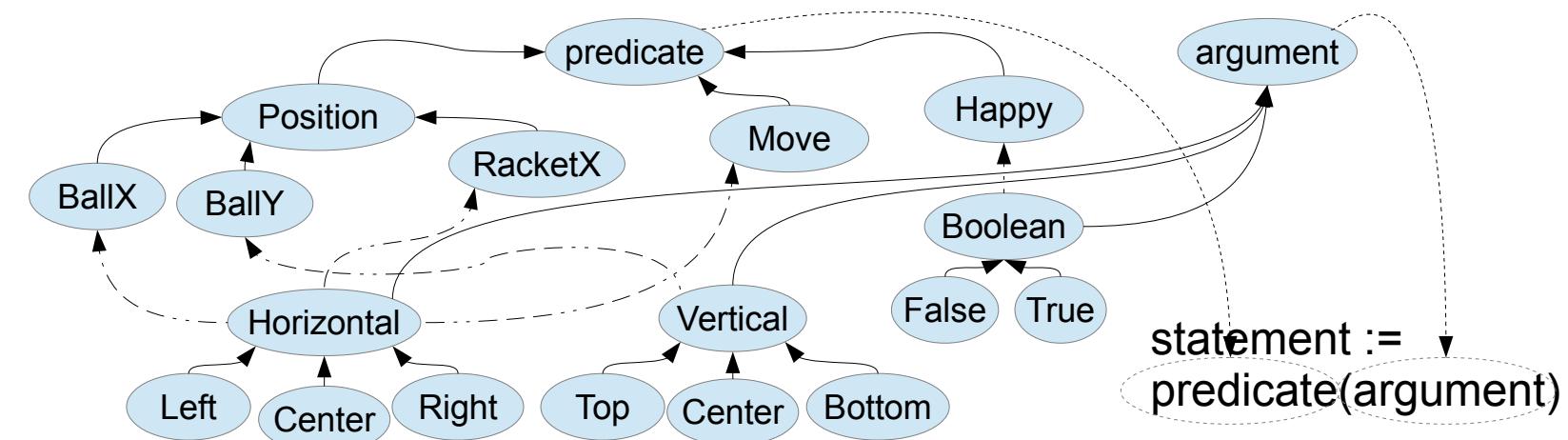
<https://www.springerprofessional.de/neuro-symbolic-architecture-for-experiential-learning-in-discret/20008336>

<https://github.com/aigents/aigents-java/tree/master/src/main/java/net/webstructor/agj>

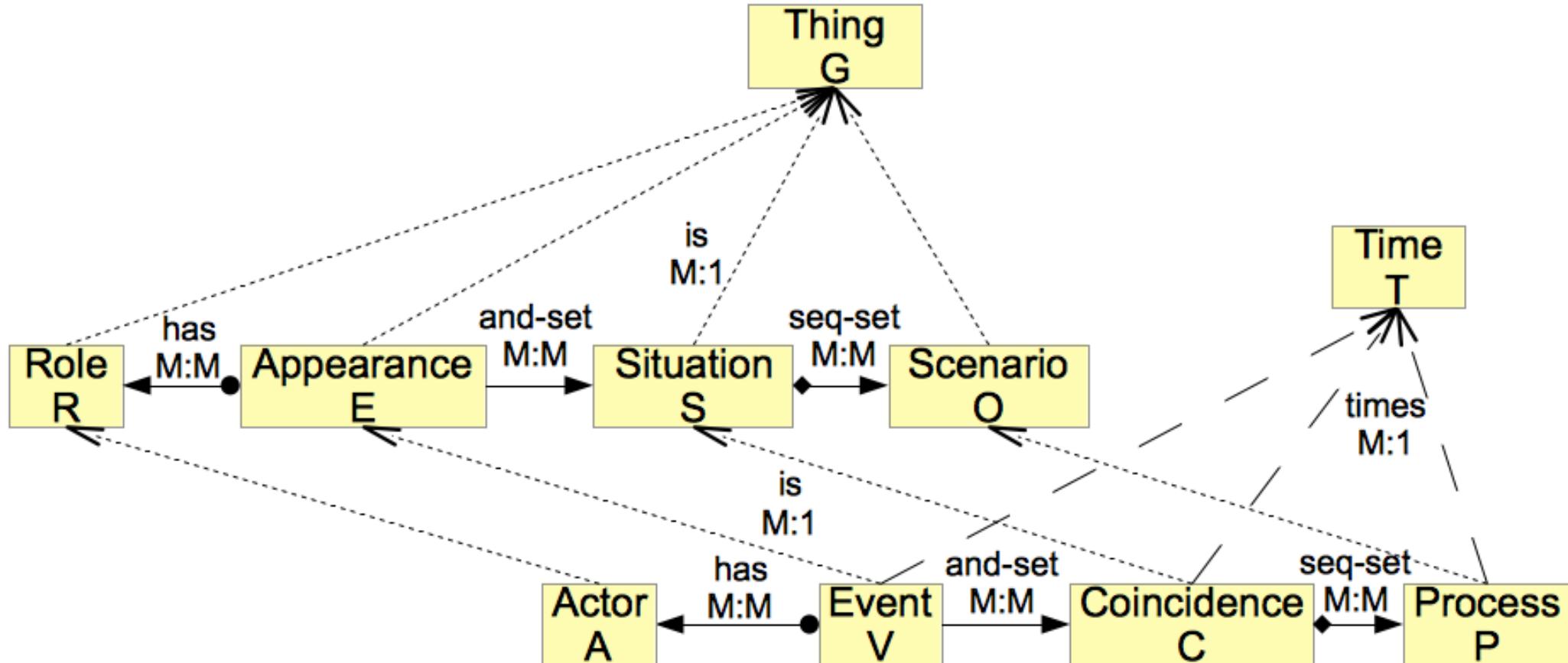
Ontology and Grammar (“Discrete”)



Ontology and Grammar (“Functional”)

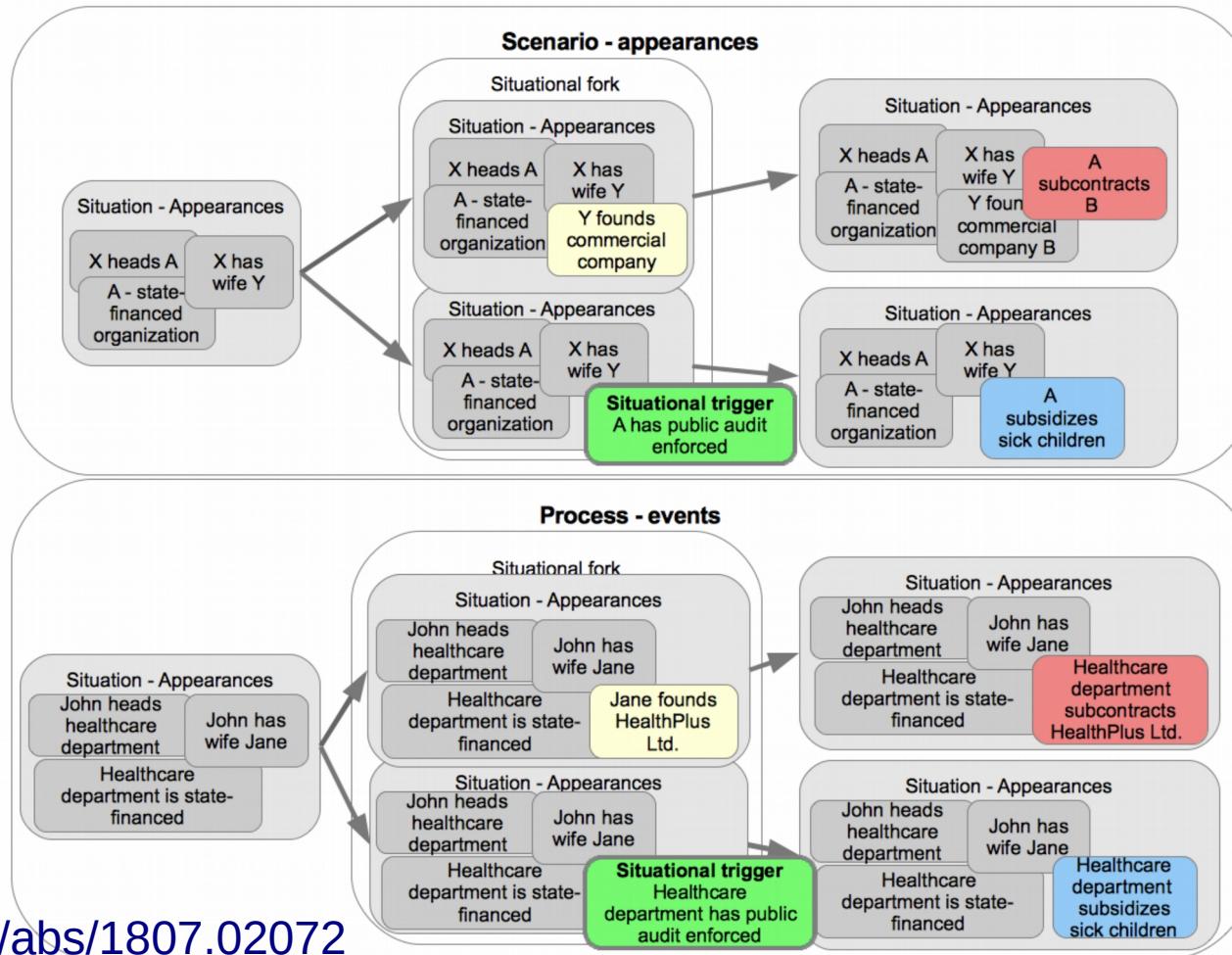


Foundation (Generic) Activity Ontology



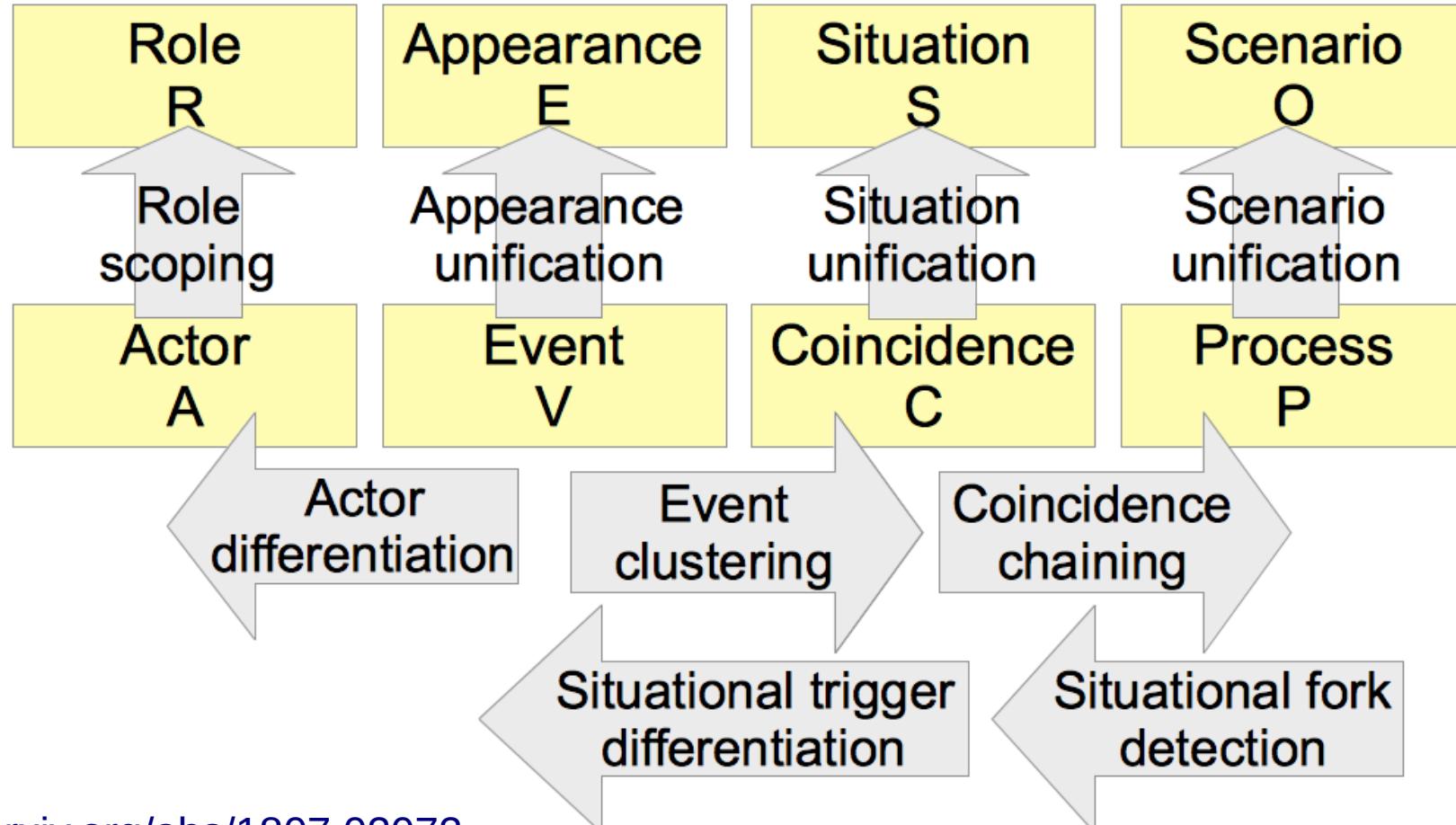
<https://arxiv.org/abs/1807.02072>

Upper (Domain) Activity Ontology Example



<https://arxiv.org/abs/1807.02072>

Scenario Mining in Process Data



<https://arxiv.org/abs/1807.02072>

Applications other than NLP and Experiential/Reinforcement Learning

Everything is a
Process of a
Scenario!

<https://arxiv.org/abs/1807.02072>

	Spoken language recognition	Written language recognition	Identification of patterns in the text	Causal analysis
Actor	Coefficient on spectrogram for particular frequency	Property of specific stroke: On the top, at the bottom, long, short, skewed, etc.	Object property value: Name “John”	Specific actor: John Doe
Event	Combination of coefficients on spectrogram	Period or stroke composing the letter: .	Object class instance: Name “John”, surname “Doe”	Specific event: John Doe cleaning the window on the second floor.
Coincidence	Specific sound	Coincidence: i	Co-occurrence of object of class person properties: “John Doe”	Specific coincidence: Window on the second floor is dirty, John Doe is cleaning it.
Process	Specific spoken word	Specific written word: ping	Specific phrase: “John Doe cleans the window”	Specific process: Window on the second floor was dirty, John Doe has cleaned it and now it is clean.
Role	Pitch frequency	Property of symbol: orientation, extent, symmetry, etc.	Domain of the object class property: person’s surname	Typical role: Cleaner
Appearance	Spectral cluster on the spectrogram	Element of symbol: .	Class of the variable object: person	Typical appearance: Someone cleaning something
Situation	Sound of speech	Symbol: i	Pattern variable: \$subject	Typical situation: Someone is cleaning something which is dirty.
Scenario	Spoken word accordingly to the language model	Written word: ping	Phrase pattern: “\$subject cleans \$object”	Typical scenario: Something was dirty, someone has cleaned it, it is clean now.

Evgeni Vityaev's Invariant Structures

Mining temporally invariant structures (“scenarios”) in historical time series data recording financial “processes” with the **Discovery** system.

http://www.math.nsc.ru/AP/ScientificDiscovery/PDF/data_mining_for_financial_applications.pdf

**Symbolic Methodology in Numeric Data Mining:
Relational Techniques for Financial Applications**
Boris Kovalerchuk, Evgenii Vityaev, Husan Yusupov
<https://arxiv.org/pdf/cs/0208022.pdf>

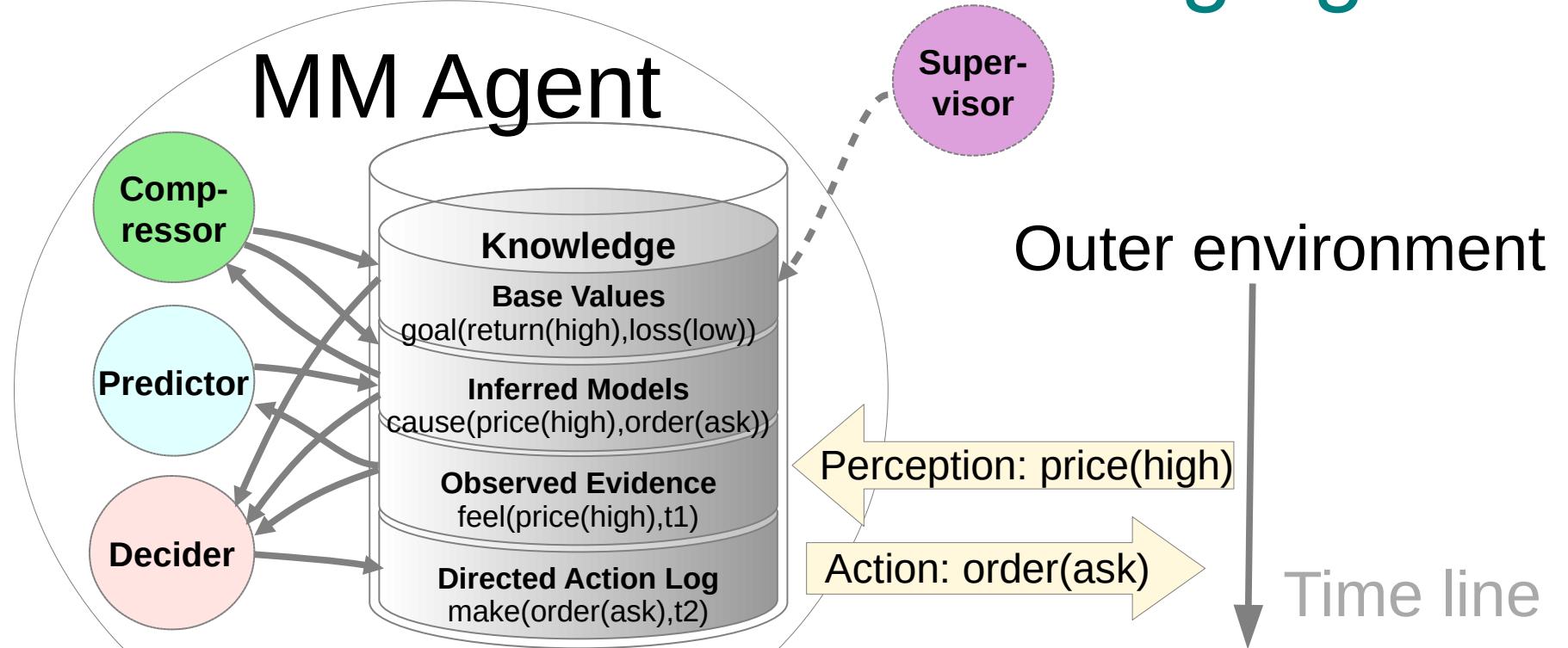
Brain Principles Programming
Evgenii Vityaev, Anton Kolonin, Andrey Kurpatov,
Artem Molchanov

<https://arxiv.org/abs/2202.12710>

ANNEX IN FINANCIAL FORECASTING

structure1	structure2	structure3	structure4	weekday	week
			forecast for	Friday	forecast week
			up	Thursday	forecast week
		forecast for	up	Wednesday	forecast week
	forecast for	up		Tuesday	forecast week
	up			Monday	forecast week
current day			forecast for	Friday	current week
	up		up	Thursday	current week
		current day		Wednesday	current week
		down		Tuesday	current week
	down			Monday	current week
		anchor2	down	Friday	one week ago
		down_anchor1		Thursday	one week ago
			down	Wednesday	one week ago
				Tuesday	one week ago
		anchor1		Monday	one week ago
			up	Friday	two weeks ago
				Thursday	two weeks ago
		anchor2		Wednesday	two weeks ago
		up		Tuesday	two weeks ago
		anchor1		Monday	two weeks ago
			up	Friday	three weeks ago
				Thursday	three weeks ago
		anchor1		Wednesday	three weeks ago
			up	Tuesday	three weeks ago
				Monday	three weeks ago
training 0.74	training 0.72	training 0.7	training 0.7		
testing 0.78	testing 0.73	testing 0.71	testing 0.82		

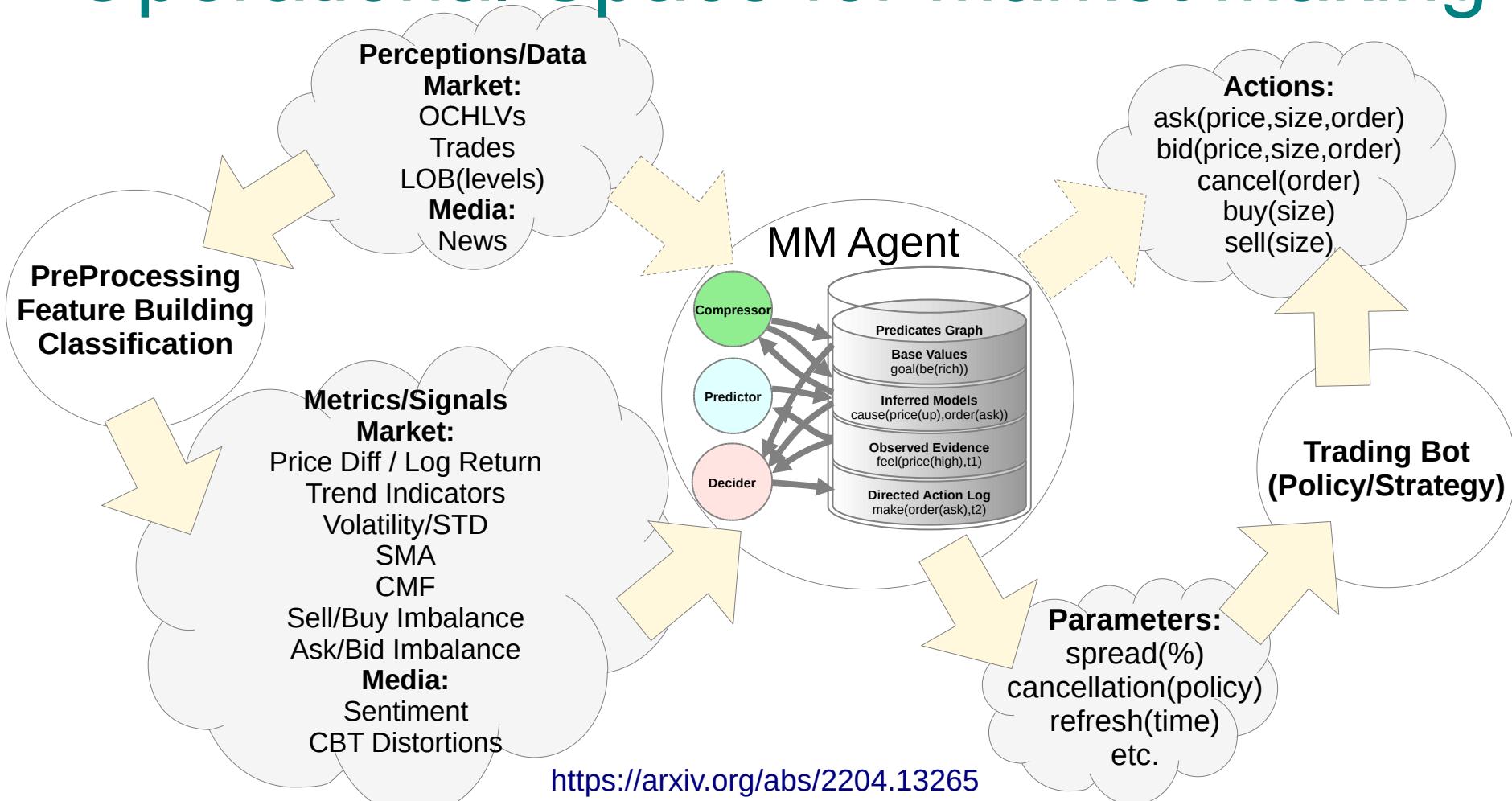
Narrow AGI for Market Making Agent



Evgenii E. Vityaev Purposefulness as a Principle of Brain Activity // Anticipation: Learning from the Past, (ed.) M. Nadin. Cognitive Systems Monographs, V.25, Chapter No.: 13. Springer, 2015, pp. 231-254.

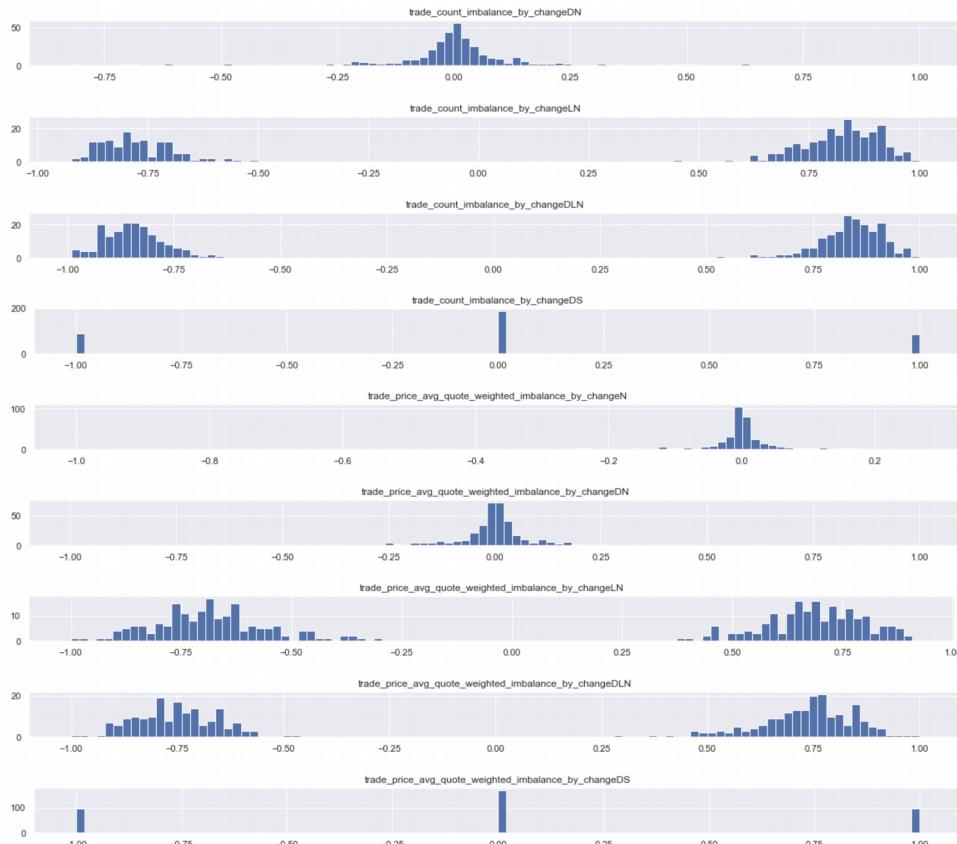
Anton Kolonin: Neuro-symbolic architecture for experiential learning in discrete and functional environments // AGI-2021 Conference Proceedings, 2021

Operational Space for Market Making

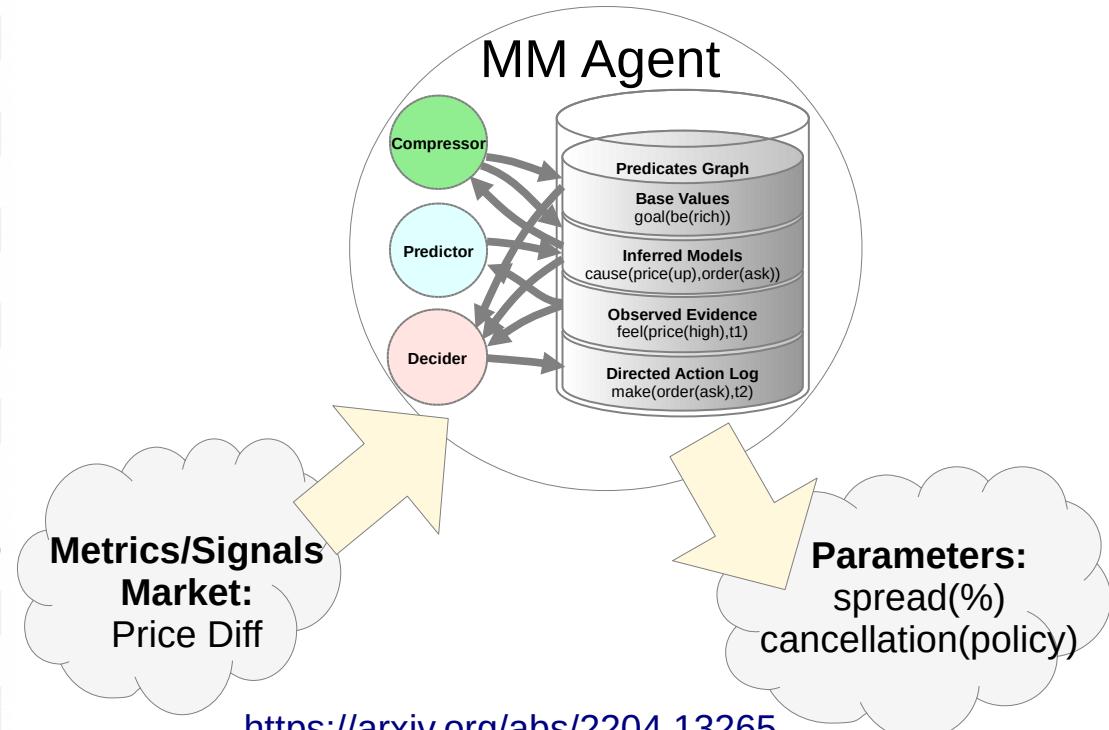


Operational Space for Market Making

More Metrics



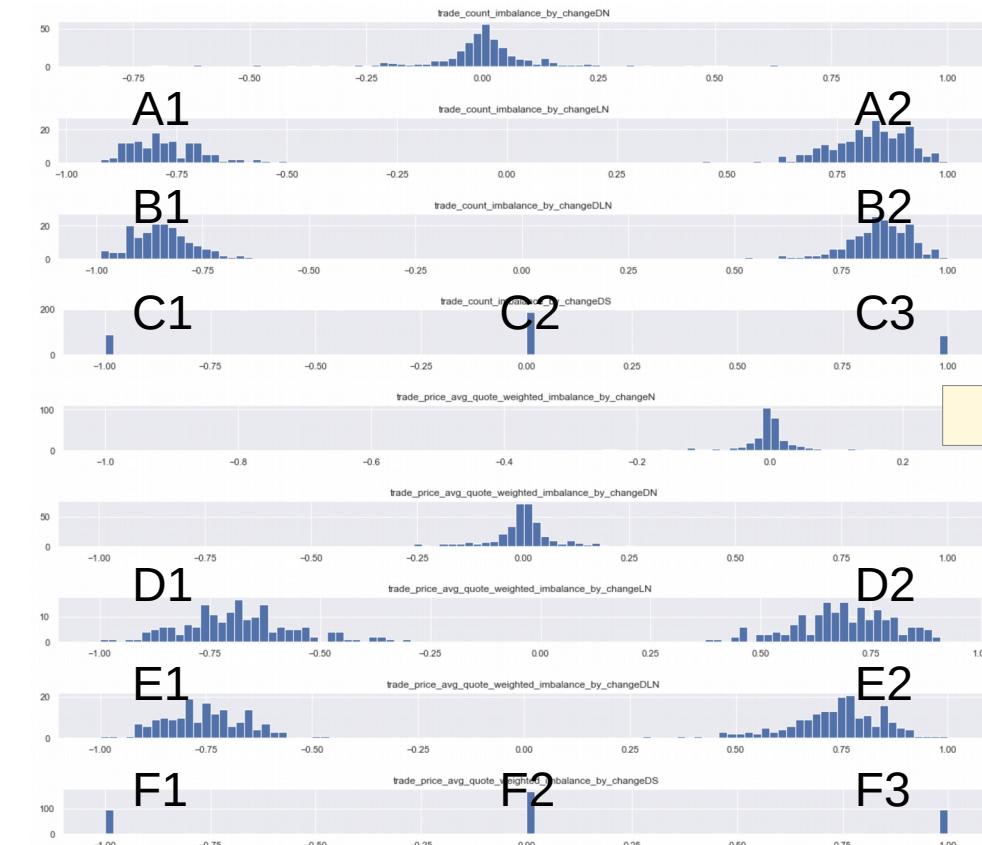
Simplified



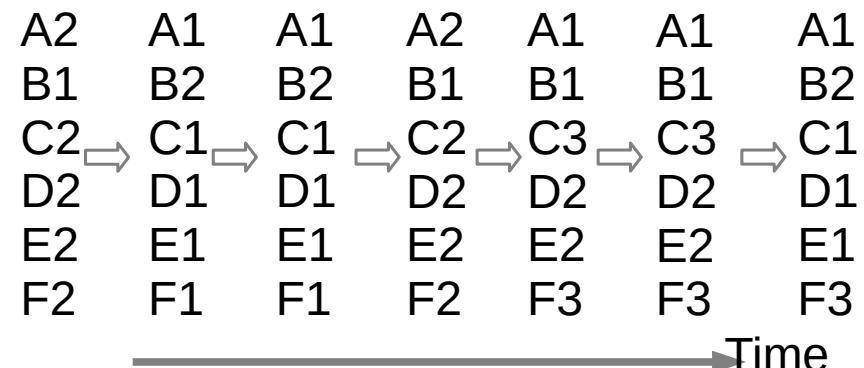
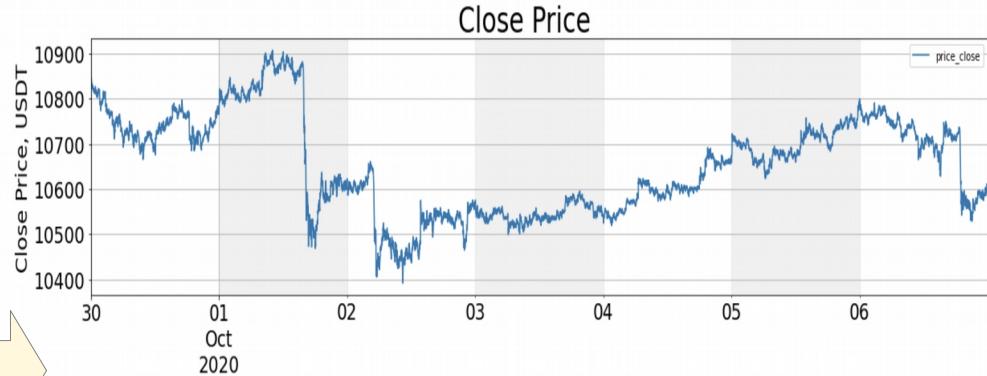
<https://arxiv.org/abs/2204.13265>

Symbolizing Market Conditions?

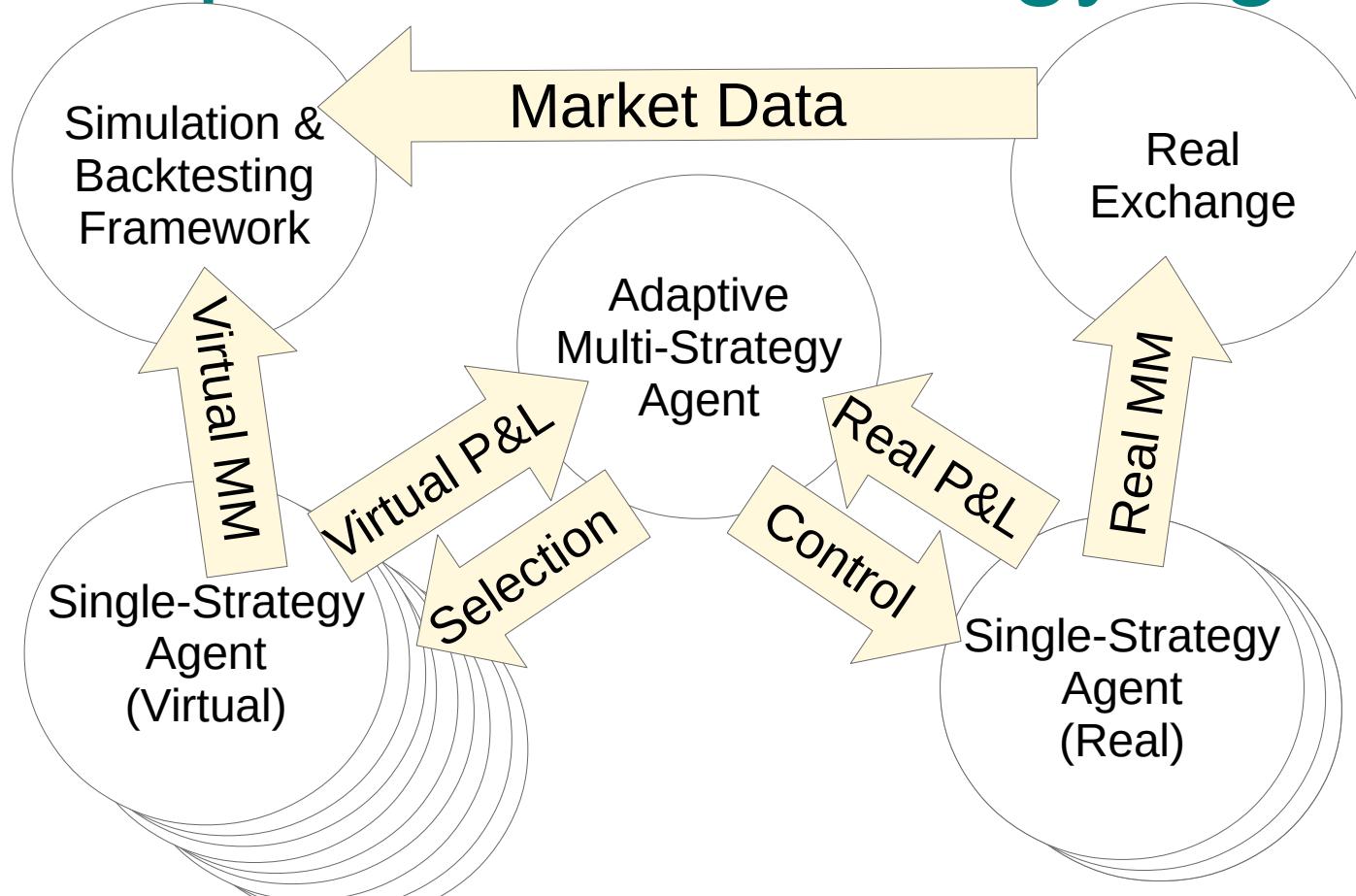
More Metrics



Flat Crash Crash Flat Up Up Crash



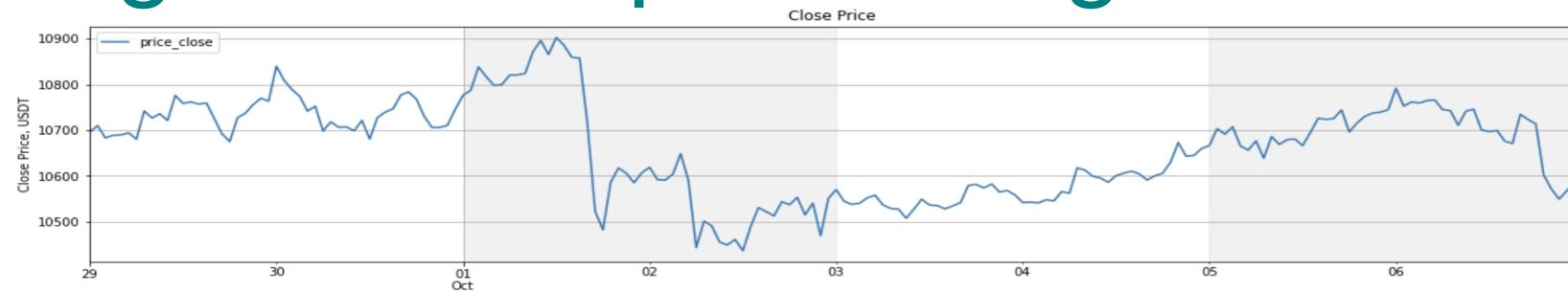
Adaptive Multi-Strategy Agent



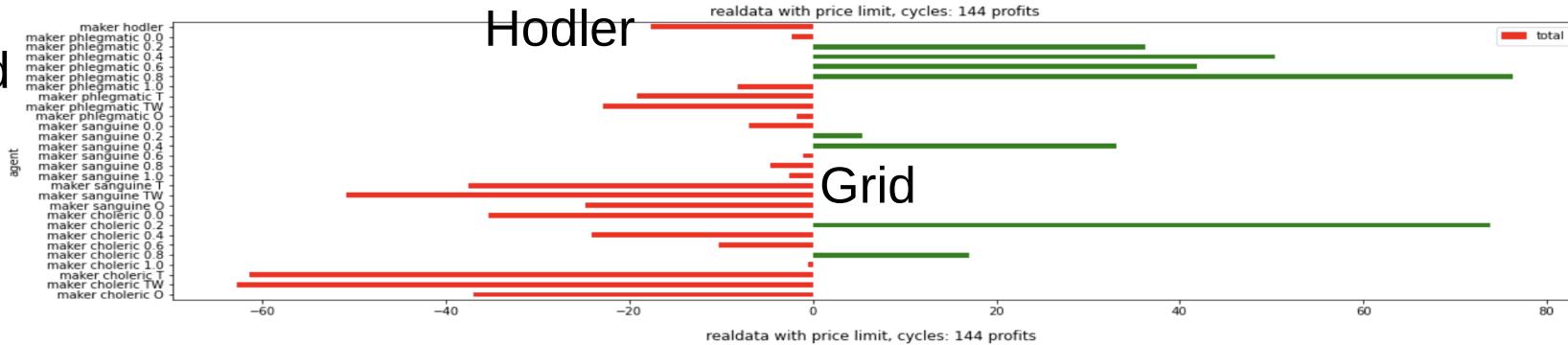
<https://arxiv.org/abs/2107.07769>

Trading with multiple Strategies at a Time

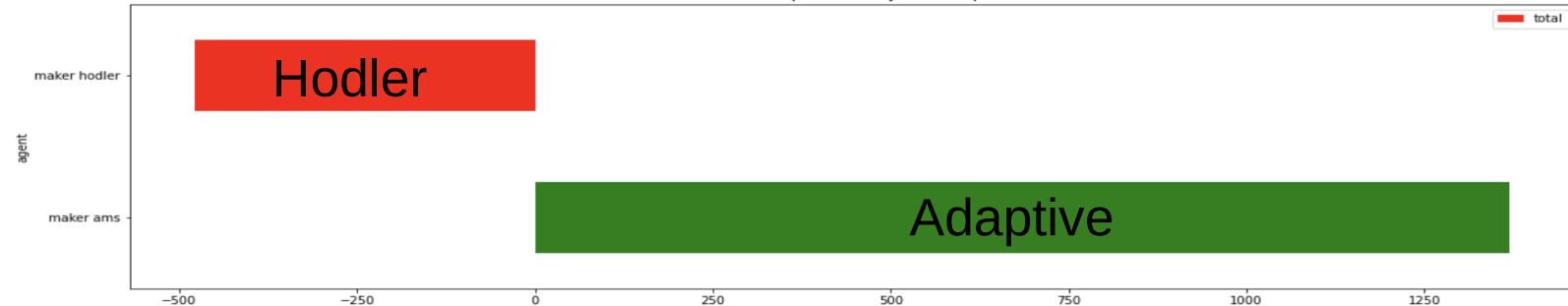
BTC/
USDT



Order Grid
Marker
Making
P & L



Adaptive
Multi
Strategy
Agent
P & L

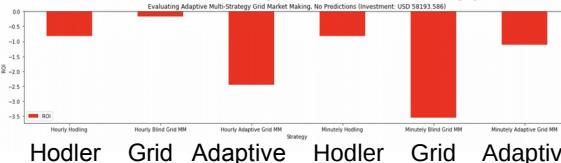


Overall ROI for adaptive MM strategy

BTC/
USDT



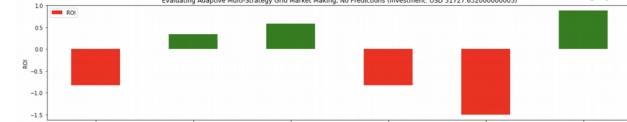
“Basic/Simple” Strategy



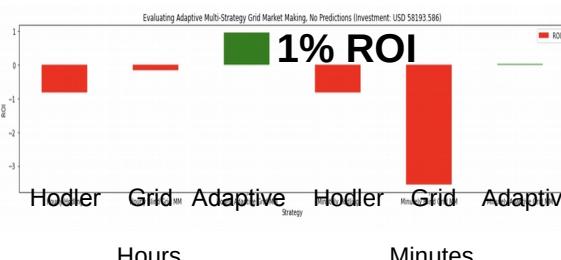
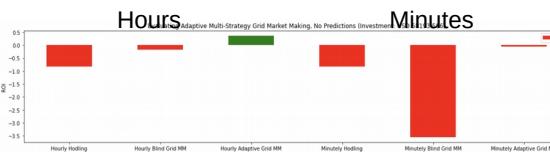
“NIOX Maker” Strategy



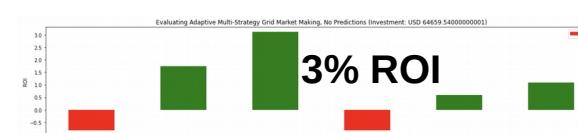
“Hummingbot Pure MM” Strategy



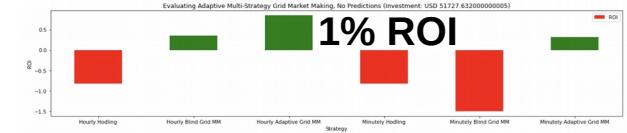
1 day strategy update interval (6 days)



2 days strategy update interval (6 days)



3 days strategy update interval (6 days)

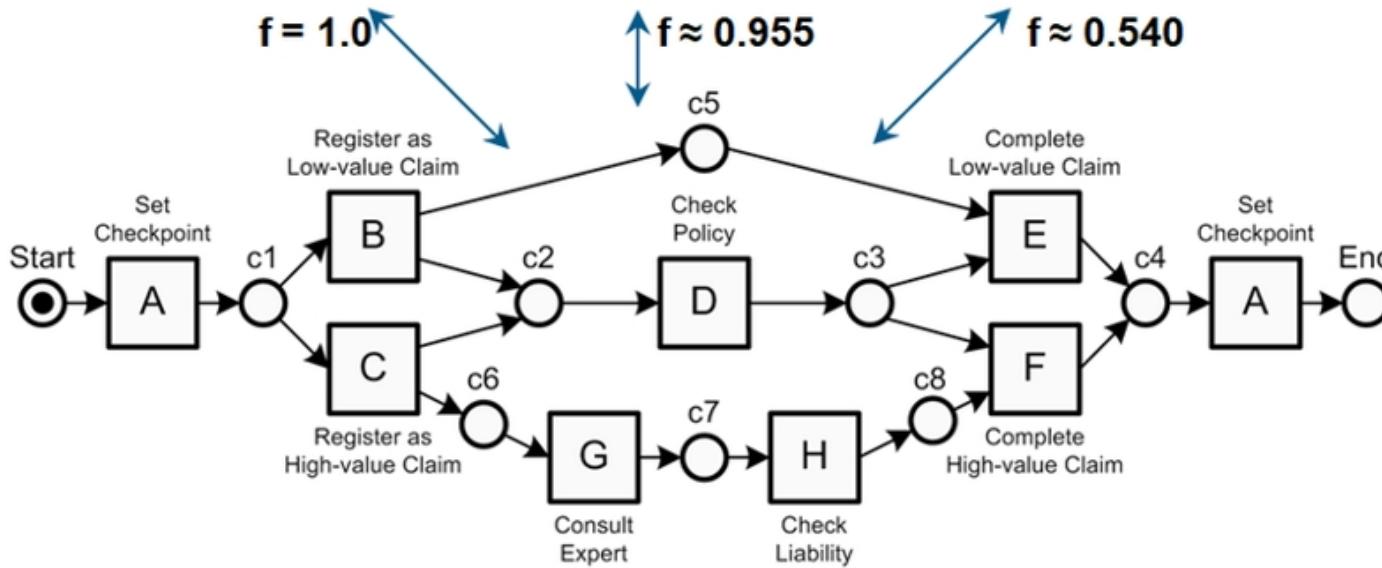


Process Mining in Business Log Data

No. of Instances	Log Traces
4070	ABDEA
245	ACDHFA
56	ACGDHFA

No. of Instances	Log Traces
1207	ABDEA
145	ACDHFA
56	ACGDHFA
23	ACHDFA
28	ACDHFA

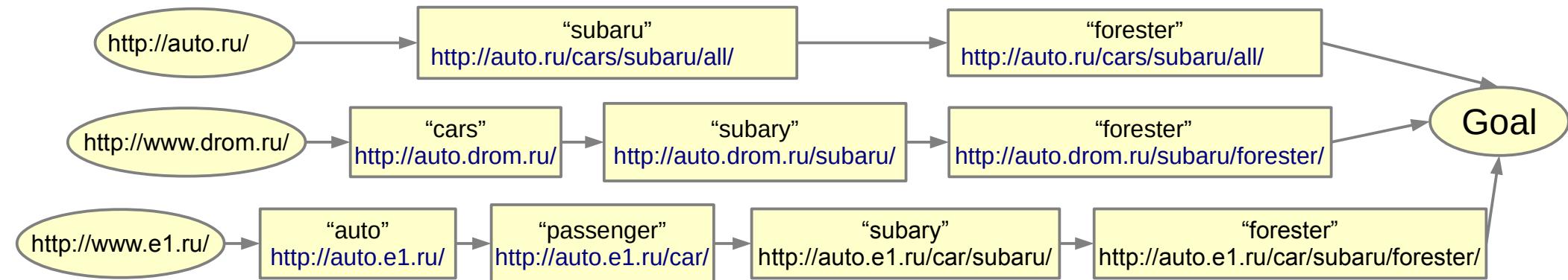
No. of Instances	Log Traces
24	BDE
7	AABHF
15	CHF
6	ADBE
1	ACBGDFAA
8	ABEDA



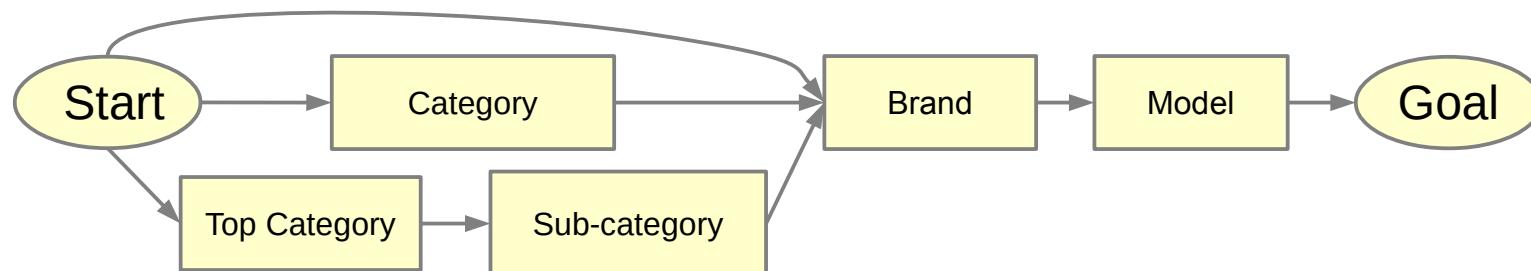
https://www.puzzledata.com/process-mining_eng/

Scenario Mining in Process Data Example

Internet navigation routes when searching for “Forester” cars



Generalized click-through scenario when searching for products

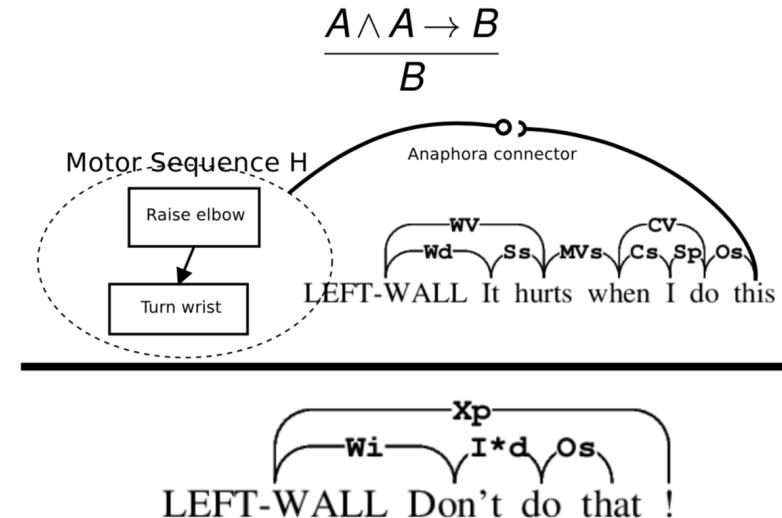


Text Grounding on Non-Text Events

Common Sense Reasoning

Everything is a
Graph and can
be framed with
Link Grammar!

Rules, laws, axioms of reasoning and inference can be learned.



Linas Vepstas, 2021

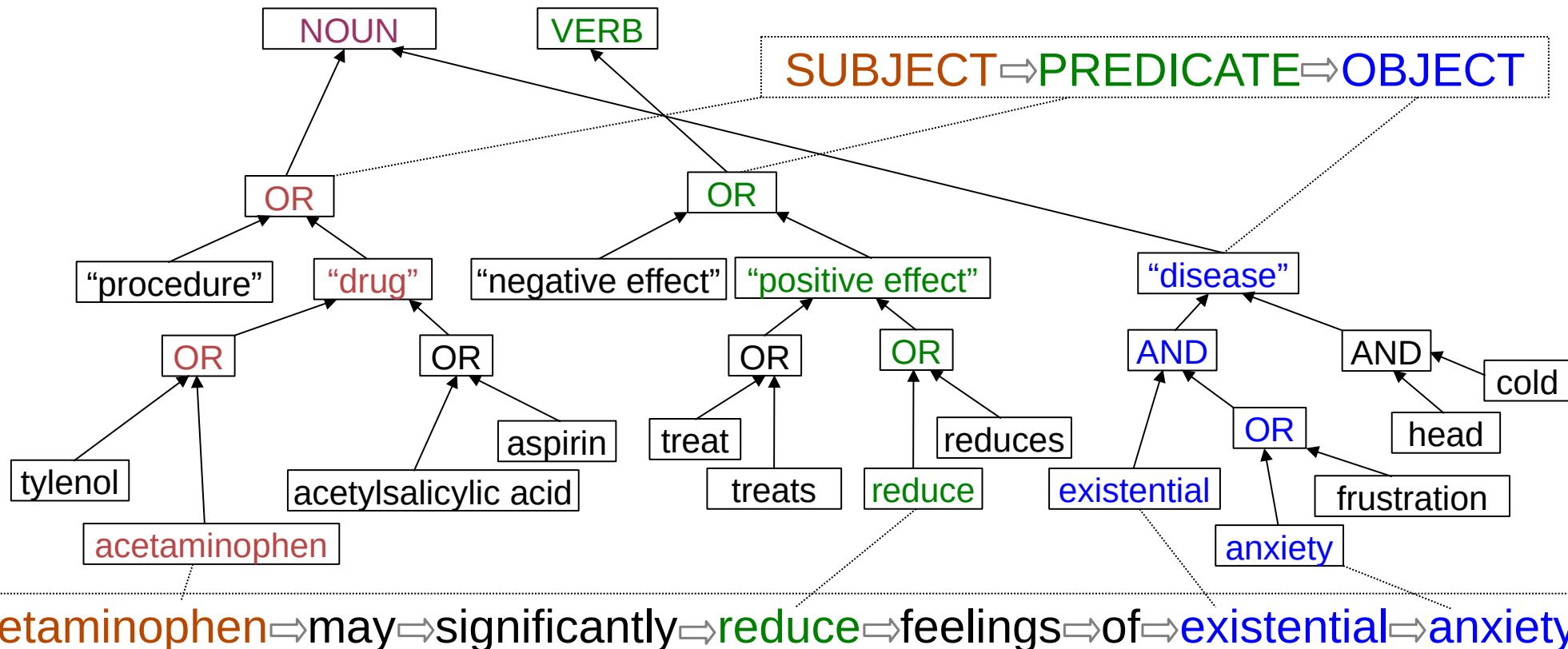
<https://arxiv.org/abs/1901.01341>

<https://aigents.github.io/inlp/2021/slides/>

Naively, simplistically: Learned Stimulus-Response AI (SRAI)⁹

⁹Metaphorical example: Mel'čuk's Meaning Text Theory (MTT) SemR + Lexical Functions (LF) would be better.

Discovering NLP patterns such as words or phrase structures for unsupervised language learning (Aigents® “Deep Patterns”)



<https://ieeexplore.ieee.org/document/7361868>
<https://github.com/aigents/aigents-java>

<https://www.springerprofessional.de/unsupervised-language-learning-in-opencog/15995030>
<https://www.springerprofessional.de/en/programmatic-link-grammar-induction-for-unsupervised-language-le/17020348>
<https://github.com/singnet/language-learning/>

Issues to Address

Absence of explicit start/stop tags in continuous streams of spaces in experiential (reinforcement/self-reinforcement) learning with delayed/sparse feedback

<https://www.youtube.com/watch?v=2LPLhJKh95g>

<https://www.springerprofessional.de/neuro-symbolic-architecture-for-experiential-learning-in-discret/20008336>

<https://github.com/aigents/aigents-java/tree/master/src/main/java/net/webstructor/agi>

Complex, cumbersome, unreliable and expensive language-specific tokenization process for unsupervised language learning in NLP

Low quality of unsupervised parsing and tokenization learning based on mutual information and conditional probabilities

<https://www.springerprofessional.de/unsupervised-language-learning-in-opencog/15995030>

<https://www.springerprofessional.de/en/programmatic-link-grammar-induction-for-unsupervised-language-le/17020348>

<https://github.com/singnet/language-learning/>

<https://scholarsarchive.byu.edu/cgi/viewcontent.cgi?article=6983&context=etd>

Tokenization or Text Segmentation as Language Modeling

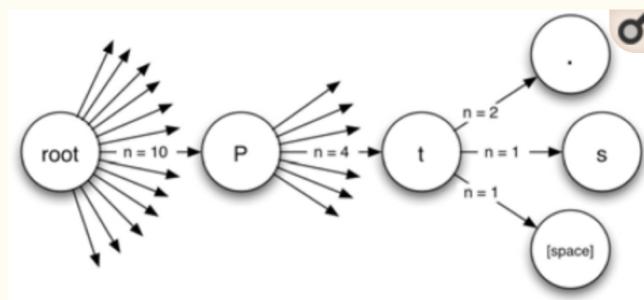


Figure 1

Trie data structure. The probability of observing an ‘s’ given the preceding string “Pt” is $\frac{1}{4}$, or 25%. The freedom following “pt” is 3.

Metrics/Indicators:

Mutual Information¹

Conditional Probability^{1,2}

Transition Freedom^{2,3}

¹ <https://scholarsarchive.byu.edu/cgi/viewcontent.cgi?article=6983&context=etd>

² <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2655800/>

³ Karl Friston. The free-energy principle: a unified brain theory?
<https://www.nature.com/articles/nrn2787>

Contrastive Evaluation: Test Specific Phenomena

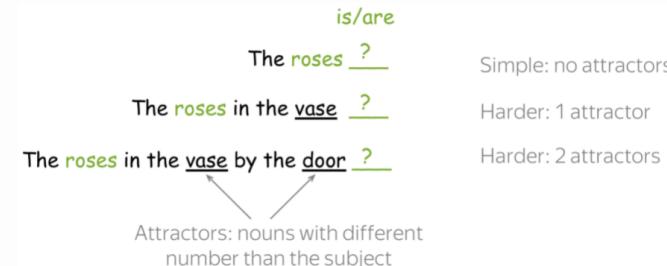
To test if your LM knows something very specific, you can use contrastive examples. These are the examples where you have several versions of the same text which differ only in the aspect you care about: one correct and at least one incorrect. A model has to assign higher scores (probabilities) to the correct version.

The roses in the vase by the door ? Competing answers: is, are

P(The roses in the vase by the door are) → Is the correct answer ranked higher?
P(The roses in the vase by the door is) → $P(\dots\text{are}) > P(\dots\text{is})$?

A very popular phenomenon to look at is subject-verb agreement, initially proposed in the [Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies](#) paper. In this task, contrastive examples consist of two sentences: one where the verb agrees in number with the subject, and another with the same verb, but incorrect inflection.

Examples can be of different complexity depending on the number of attractors: other nouns in a sentence that have different grammatical number and can “distract” a model from the subject.



https://lena-voita.github.io/nlp_course/language_modeling.html

Claims

Transition Freedom (TF) appears to be superior (over **Mutual Information** and **Conditional Probability**) for unsupervised text segmentation (tokenization).

English and Russian require one specific way (variance) of handling the TF while Chinese requires a bit different specific way (derivative-based “peak values”) for the same purpose.

Tokenization quality for Russian and English may be as high as $F1=0.96-1.0$, depending on training and testing corpora while for Chinese the minimum is $F1=0.71-0.92$, depending on the assessment assumptions.

Larger training corpora does not necessarily effect in better tokenization quality, while compacting the models eliminating statistically weak evidence typically improve the quality.

TF-based tokenization appear quality same or better than lexicon-based one for Russian and English while for Chinese appears the opposite (as it could be anticipated).

Doing Russian and English tokenization with removed spaces makes the situation similar to Chinese with reasonable quality on lexicon-based tokenization but much worse results on TF-based one.

<https://arxiv.org/abs/2205.11443>

<https://github.com/aigents/pygents>

Corpora and Methodology

Train corpora

Chinese

CLUE News 2016 Validation – 270M

CLUE News 2016 Train – 8,500M

English

Brown – 6M

Gutenberg Children – 29M

Gutenberg Adult – 140M

Social Media – 68M

All above – combined

Russian

RusAge Test – 141M

RusAge Previews – 825M

Test corpus

Parallel Chinese/English/Russian

– 100 multi-sentence statements on finance

Metrics/Indicators:

Ngram (Character)

Probability or Conditional Transition Probability (p-/p+)

Deviation (dvp-/dvp+)

Derivative (dp-/dp+)

Transition Freedom (f-/f+)

Deviation (dvf-/dvf+)

Derivative (df-/df+)

Hyper-parameters:

Combination of Ngram ranks N ([1],[2],[3],[1,2],[1,2,3],...)

Threshold for model compression

Threshold for segmentation

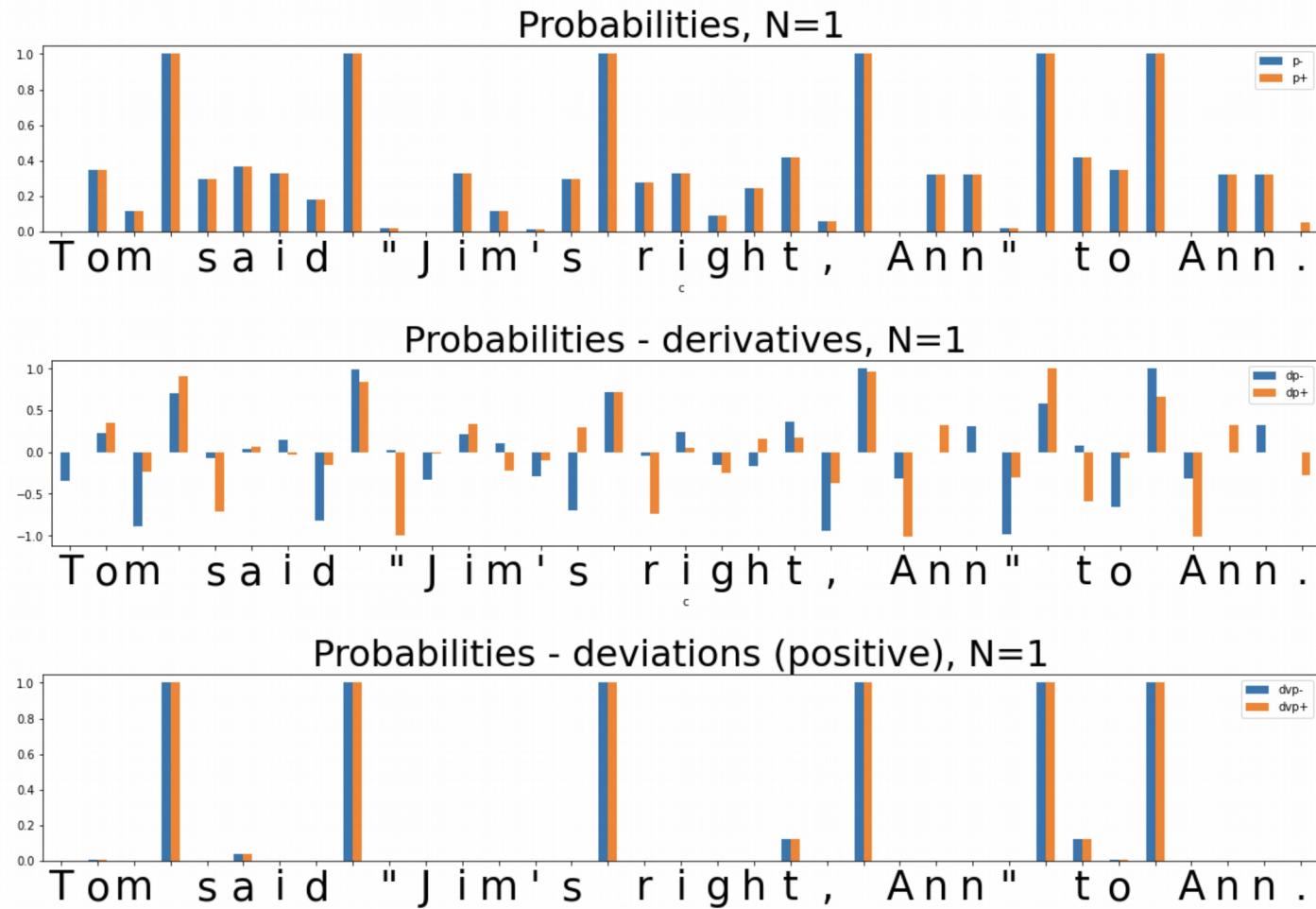
Evaluations:

Tokenization F1, on set of tokens found comparing to delimiter-based (English/Russian) or Jieba (Chinese)

Precision on set of tokens found comparing to reference lexicons

Unsupervised Text Segmentation (Tokenization)

Metrics/Indicators:
Ngram (Character)
Probability



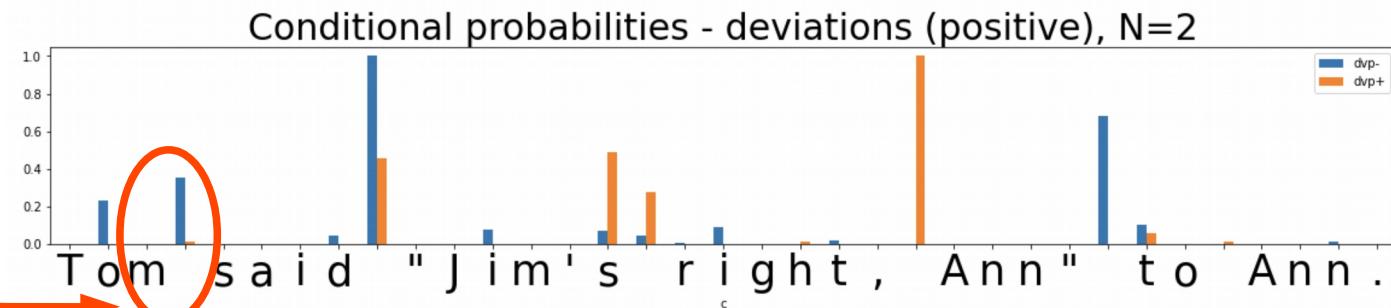
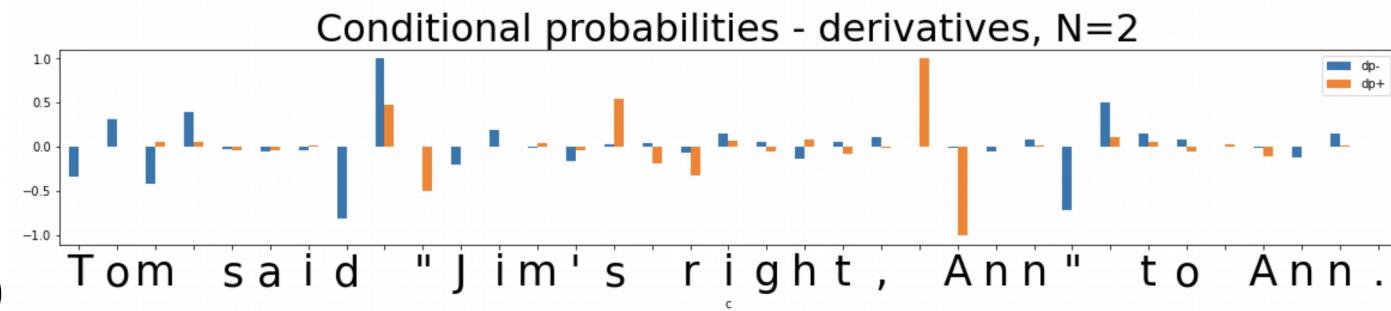
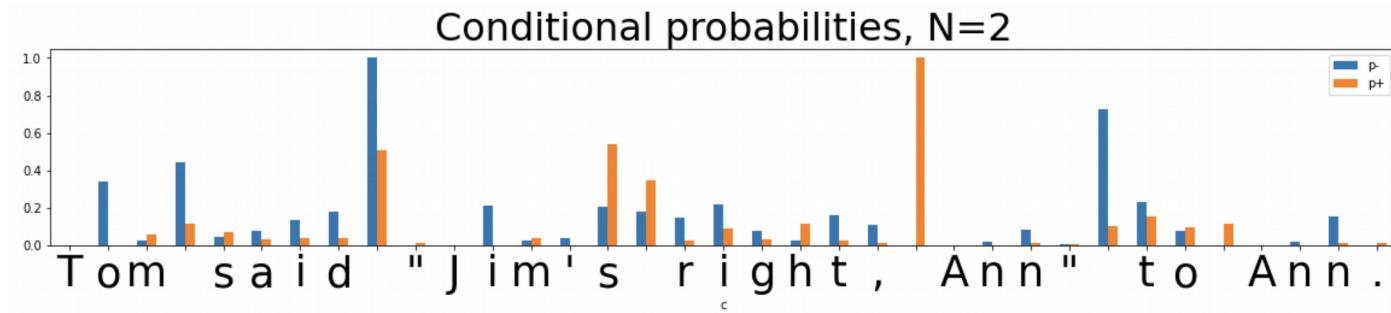
Unsupervised Text Segmentation (Tokenization)

Metrics/Indicators:

Ngram (Character)
Conditional
Probability
(of Transition)

$P(\text{Ngram}_{n+1})/P(\text{Ngram}_n)$

$P("m")/P(m")$



Unsupervised Text Segmentation (Tokenization)

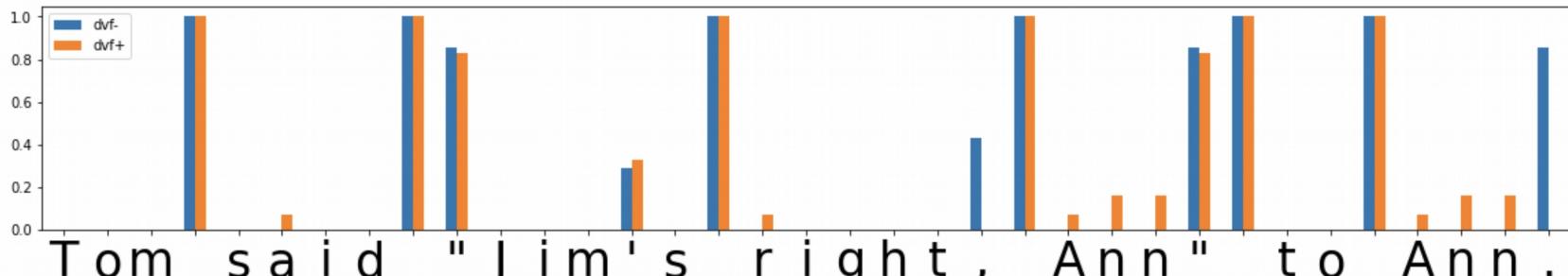
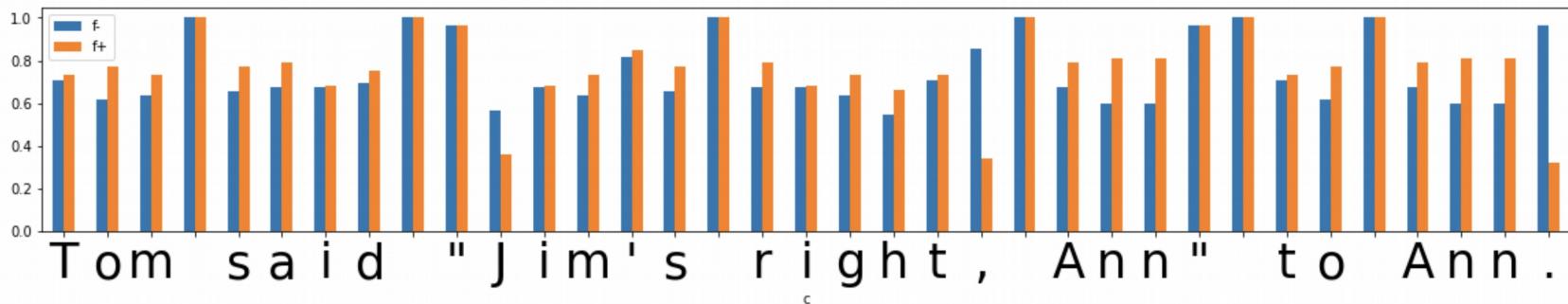
```
Threshold 0.25
Tom said "Jim's right, Ann" to Ann.
['Tom', ' ', 'said', ' ', "'", "Jim's", ' ', 'right', ' ', ' ', 'Ann', ' ', ' ', 'to', ' ', ' ', 'Ann', '.']
['Tom', ' ', 'said', ' ', "'", "Jim", " ", 's', ' ', 'right', ' ', ' ', 'Ann', ' ', ' ', 'to', ' ', ' ', 'Ann', '.']
0.89
```

```
Threshold 0.35
Tom said "Jim's right, Ann" to Ann.
['Tom', ' ', 'said', ' ', "'", "Jim's", ' ', 'right', ' ', ' ', 'Ann', ' ', ' ', 'to', ' ', ' ', 'Ann', '.']
['Tom', ' ', 'said', ' ', "'", "Jim's", ' ', 'right', ' ', ' ', 'Ann', ' ', ' ', 'to', ' ', ' ', 'Ann', '.']
1.0
```

Metrics/ Indicators:

Transition
Freedom
Deviation

(Freedom
of Transition)



Unsupervised Text Segmentation (Tokenization)

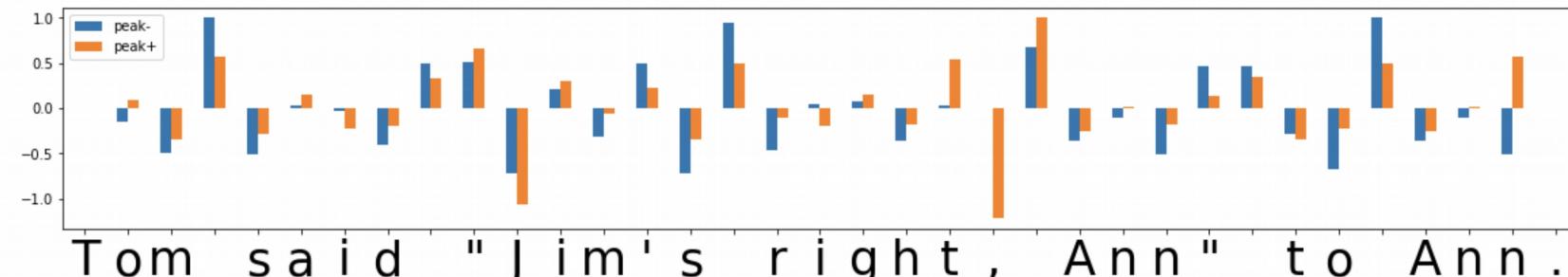
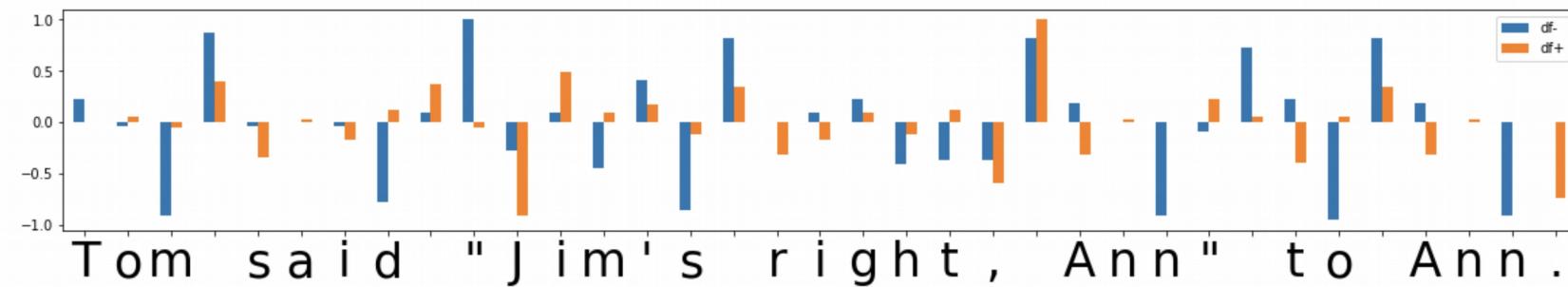
Metrics/
Indicators:

Transition
Freedom
Derivative
and “Peak”

(Freedom
of Transition
“Peak”)

Threshold 0.25
Tom said "Jim's right, Ann" to Ann.
['Tom', ' ', 'said', ' ', "'", "Jim's", ' ', 'right', ' ', ' ', 'Ann', ' ", " ', 'to', ' ', 'Ann', '.']
['Tom', ' ', 'said', ' ', "'", 'Ji', 'm', 's', ' ', 'right', ' ', ' ', 'Ann', ' ", " ', 'to', ' ', 'Ann', '.']
0.89

Threshold 0.35
Tom said "Jim's right, Ann" to Ann.
['Tom', ' ', 'said', ' ', "'", "Jim's", ' ', 'right', ' ', ' ', 'Ann', ' ", " ', 'to', ' ', 'Ann', '.']
['Tom', ' ', 'said', ' ', "'", 'Jim', 's', ' ', 'right', ' ', ' ', 'Ann', ' ", " ', 'to', ' ', 'Ann', '.']
0.82

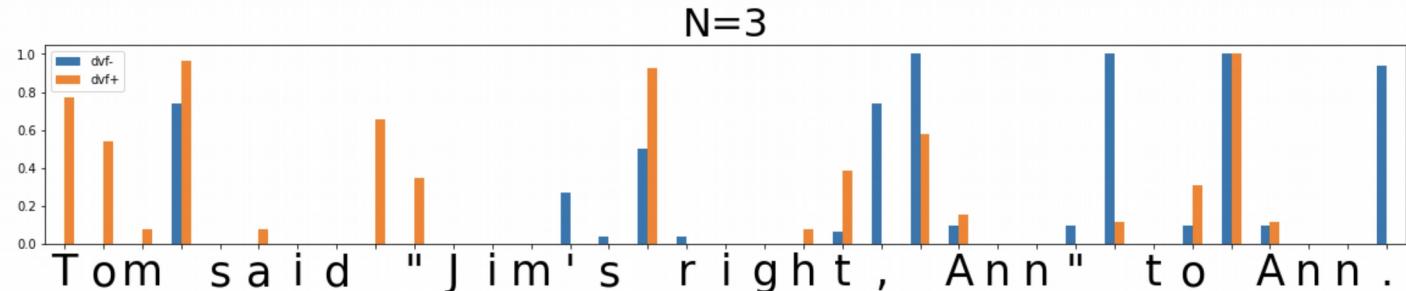
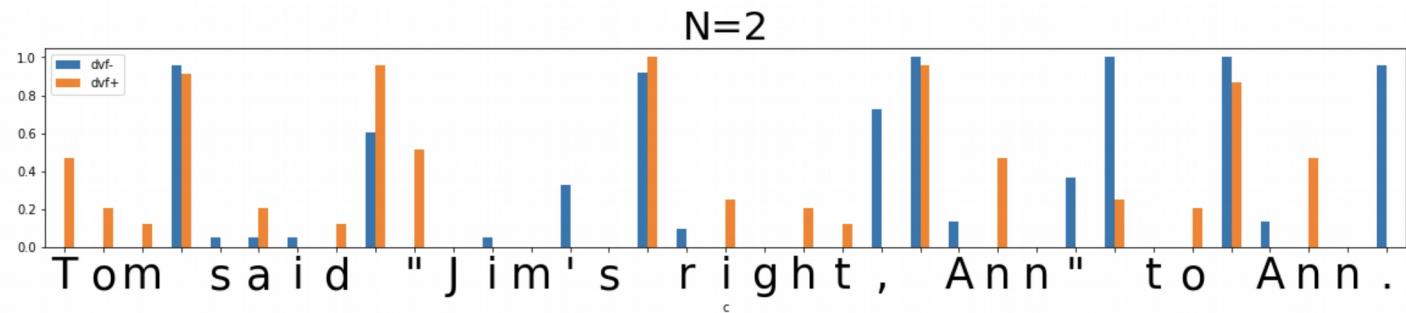
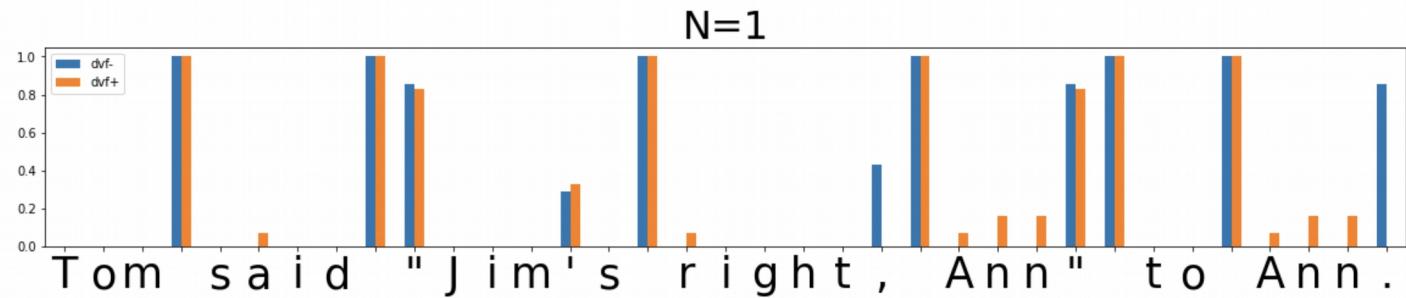


Unsupervised Text Segmentation (Tokenization)

Metrics/
Indicators:

Transition
Freedom
Deviation

(varying “N”)



Unsupervised Text Segmentation (Tokenization)

English

**Hyper-
Parameters:**

TF

**Threshold
for model
compression**

**Combination
of Ngram N-s**

**Threshold for
segmentation**

F1 - Brown ddf- & ddf+ filter=0 parameters=10967135

[1]	0.5	0.75	0.82	0.79	0.79	0.81	0.89	0.89	0.89
[2]	0.46	0.54	0.62	0.67	0.85	0.92	0.81	0.71	0.37
[3]	0.56	0.67	0.72	0.73	0.69	0.61	0.46	0.36	0.19
[4]	0.54	0.68	0.7	0.6	0.43	0.3	0.19	0.15	0.1
[5]	0.51	0.55	0.52	0.38	0.25	0.16	0.11	0.1	0.08
[6]	0.48	0.46	0.38	0.25	0.17	0.12	0.1	0.08	0.07
[7]	0.42	0.34	0.24	0.15	0.11	0.1	0.08	0.08	0.07
[1, 2]	0.47	0.58	0.82	0.94	0.94	0.91	0.89	0.79	0.56
[2, 3]	0.51	0.62	0.74	0.79	0.83	0.81	0.66	0.46	0.24
[1, 2, 3]	0.5	0.69	0.79	0.87	0.91	0.89	0.78	0.58	0.25
[1, 2, 3, 4]	0.55	0.75	0.84	0.86	0.84	0.75	0.52	0.31	0.15
[4, 5, 6, 7]	0.56	0.6	0.51	0.33	0.2	0.14	0.1	0.08	0.07
[1, 2, 3, 4, 5]	0.56	0.78	0.86	0.84	0.74	0.53	0.31	0.17	0.1
[1, 2, 3, 4, 5, 6, 7]	0.59	0.78	0.82	0.69	0.49	0.26	0.15	0.09	0.07
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9

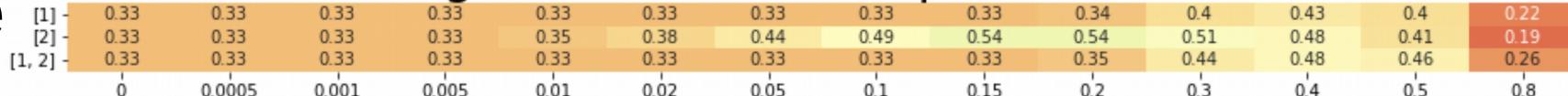
F1 - Brown ddf- & ddf+ filter=0.0001 parameters=8643703

[1]	0.73	0.96	0.98	0.99	0.96	0.94	0.95	0.95	0.89
[2]	0.46	0.54	0.64	0.79	0.91	0.94	0.89	0.7	0.44
[3]	0.55	0.66	0.74	0.78	0.72	0.65	0.49	0.37	0.19
[4]	0.54	0.67	0.7	0.61	0.45	0.32	0.21	0.16	0.1
[5]	0.51	0.55	0.52	0.38	0.26	0.17	0.12	0.1	0.08
[6]	0.48	0.46	0.38	0.26	0.18	0.13	0.1	0.09	0.07
[7]	0.42	0.35	0.25	0.16	0.12	0.1	0.09	0.08	0.08
[1, 2]	0.51	0.64	0.82	0.96	0.96	0.96	0.9	0.88	0.68
[2, 3]	0.5	0.62	0.74	0.85	0.89	0.86	0.71	0.51	0.27
[1, 2, 3]	0.53	0.69	0.81	0.91	0.93	0.92	0.82	0.6	0.36
[1, 2, 3, 4]	0.55	0.75	0.86	0.88	0.88	0.81	0.57	0.33	0.17
[4, 5, 6, 7]	0.56	0.6	0.52	0.35	0.22	0.15	0.1	0.09	0.07
[1, 2, 3, 4, 5]	0.57	0.79	0.88	0.86	0.78	0.59	0.33	0.18	0.1
[1, 2, 3, 4, 5, 6, 7]	0.59	0.79	0.83	0.71	0.5	0.28	0.16	0.09	0.08
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9

Unsupervised Text Segmentation (Tokenization)

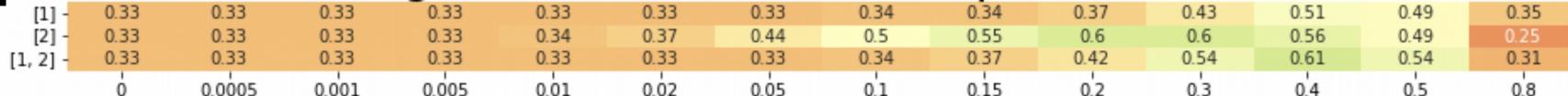
Chinese

F1 - Train(Large) f- & f+ filter=0 parameters=249859247



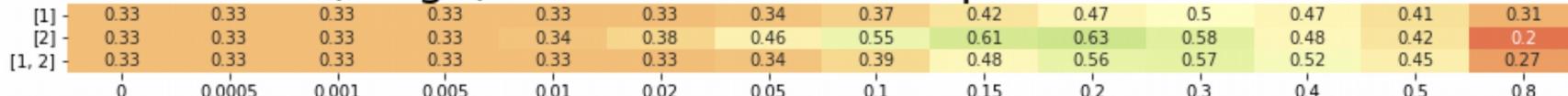
Hyper-Parameters:

F1 - Train(Large) f- & f+ filter=0.0001 parameters=231751412



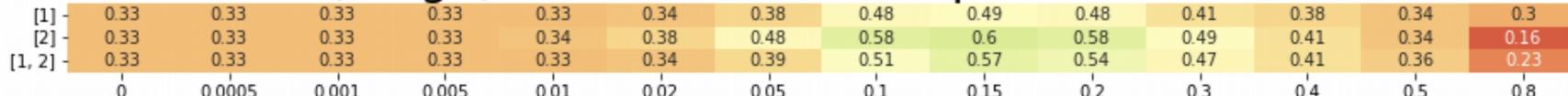
TF

F1 - Train(Large) f- & f+ filter=0.001 parameters=196866127



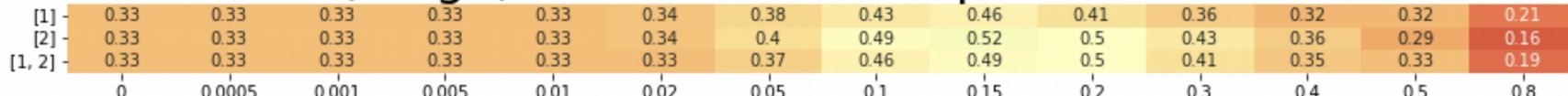
**Threshold
for model
compression**

F1 - Train(Large) f- & f+ filter=0.01 parameters=123792046



**Combination
of Ngram N-s**

F1 - Train(Large) f- & f+ filter=0.1 parameters=51791264



**Threshold for
segmentation**

Unsupervised Text Segmentation (Tokenization)

Chinese

F1 - Train(Large) peak- & peak+ filter=0 parameters=249859247

[1]	0.48	0.48	0.49	0.49	0.49	0.49	0.49	0.48	0.46	0.42	0.37	0.32	0.3	0.18
[2]	0.68	0.68	0.68	0.68	0.68	0.68	0.67	0.66	0.63	0.6	0.53	0.44	0.35	0.17
[1, 2]	0.65	0.65	0.65	0.65	0.65	0.65	0.66	0.64	0.63	0.6	0.52	0.42	0.35	0.2
	0	0.0005	0.001	0.005	0.01	0.02	0.05	0.1	0.15	0.2	0.3	0.4	0.5	0.8

Hyper-Parameters:

F1 - Train(Large) peak- & peak+ filter=0.0001 parameters=231751412

[1]	0.58	0.58	0.58	0.58	0.58	0.57	0.57	0.56	0.56	0.53	0.5	0.43	0.37	0.23
[2]	0.7	0.7	0.7	0.7	0.7	0.7	0.69	0.68	0.66	0.65	0.57	0.48	0.38	0.18
[1, 2]	0.68	0.68	0.68	0.68	0.68	0.68	0.67	0.66	0.64	0.61	0.55	0.45	0.39	0.21
	0	0.0005	0.001	0.005	0.01	0.02	0.05	0.1	0.15	0.2	0.3	0.4	0.5	0.8

TF “Peak”

Threshold
for model
compression

Combination
of Ngram N-s

Threshold for
segmentation

F1 - Train(Large) peak- & peak+ filter=0.001 parameters=196866127

[1]	0.58	0.58	0.58	0.58	0.57	0.57	0.57	0.55	0.53	0.49	0.43	0.37	0.32	0.24
[2]	0.71	0.71	0.71	0.71	0.71	0.71	0.71	0.68	0.65	0.58	0.48	0.39	0.33	0.17
[1, 2]	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.62	0.59	0.55	0.46	0.38	0.34	0.21
	0	0.0005	0.001	0.005	0.01	0.02	0.05	0.1	0.15	0.2	0.3	0.4	0.5	0.8

F1 - Train(Large) peak- & peak+ filter=0.01 parameters=123792046

[1]	0.55	0.55	0.55	0.55	0.55	0.55	0.54	0.5	0.47	0.43	0.35	0.32	0.29	0.25
[2]	0.69	0.68	0.68	0.68	0.68	0.68	0.66	0.61	0.55	0.49	0.4	0.33	0.27	0.16
[1, 2]	0.62	0.62	0.62	0.63	0.62	0.61	0.62	0.57	0.51	0.48	0.41	0.34	0.3	0.2
	0	0.0005	0.001	0.005	0.01	0.02	0.05	0.1	0.15	0.2	0.3	0.4	0.5	0.8

F1 - Train(Large) peak- & peak+ filter=0.1 parameters=51791264

[1]	0.51	0.51	0.51	0.51	0.52	0.52	0.5	0.44	0.39	0.35	0.31	0.29	0.28	0.17
[2]	0.62	0.61	0.61	0.62	0.62	0.62	0.6	0.53	0.47	0.43	0.35	0.28	0.22	0.14
[1, 2]	0.59	0.58	0.58	0.59	0.59	0.58	0.56	0.51	0.46	0.41	0.33	0.29	0.27	0.18
	0	0.0005	0.001	0.005	0.01	0.02	0.05	0.1	0.15	0.2	0.3	0.4	0.5	0.8

Unsupervised Text Segmentation (Tokenization)

The father told the mother that the child was right.

Threshold 0.15

父亲告诉母亲，孩子是对的。

['父亲', '告诉', '母亲', '，', '孩子', '是', '对', '的', '。']

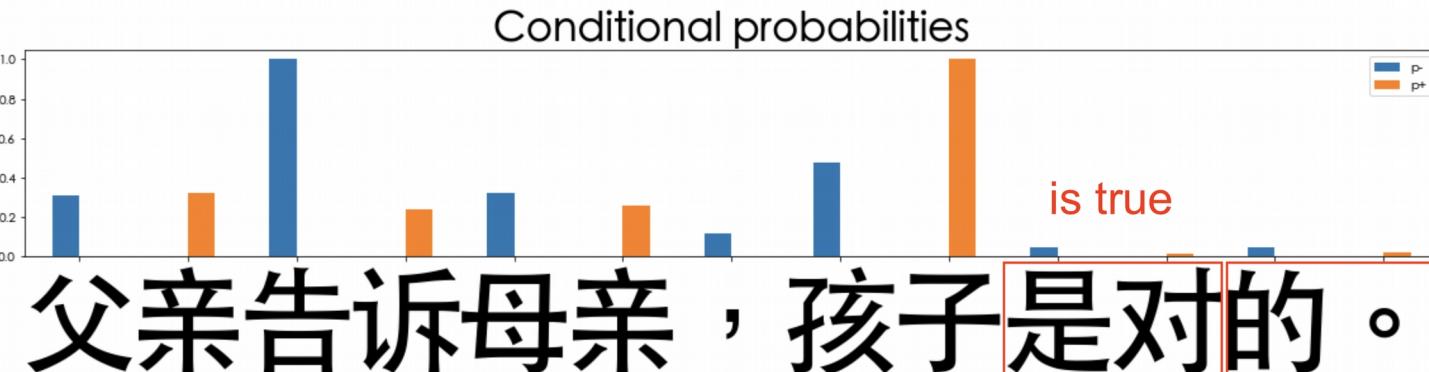
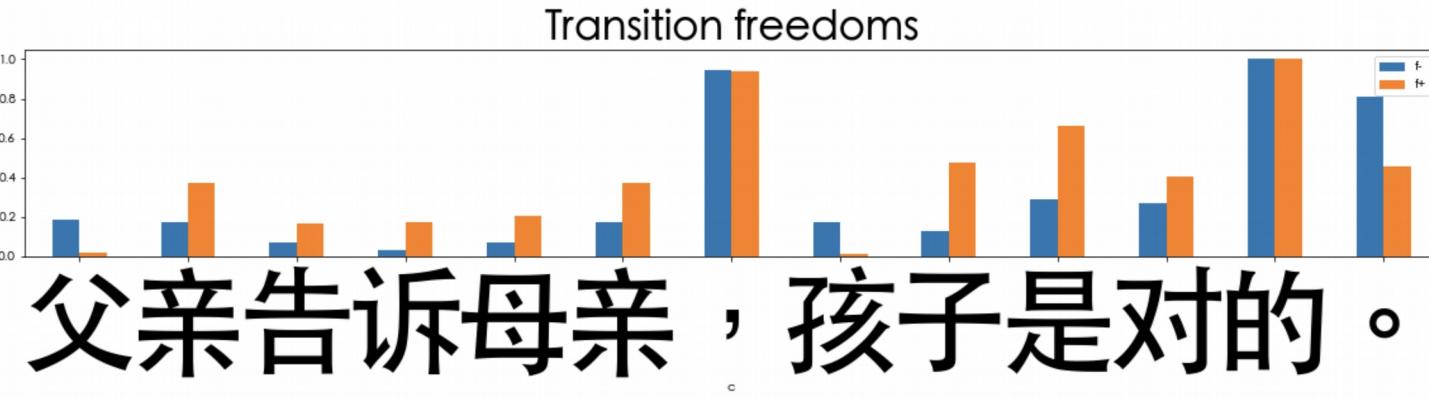
['父亲', '告诉', '母亲', '，', '孩子', '是', '对', '的', '。']

1.0

Metrics/Indicators:

Transition
Freedom
Deviation

Conditional
Probability
(of Transition)



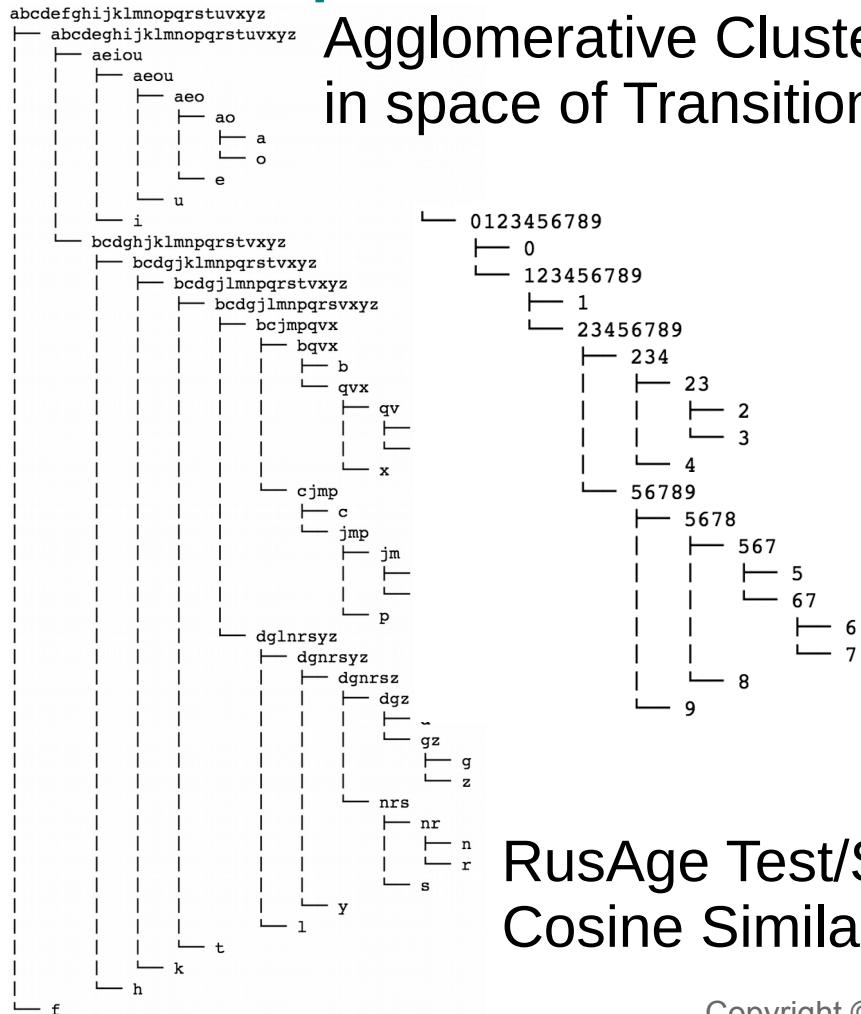
Results – Freedom-based Tokenization against Lexicon

Language	Tokenizer	Tokenization F1	Lexicon Discovery Precision
English	Freedom-based	0.99	0.99 (vs 1.0)
English	Lexicon-based	0.99	-
English no spaces	Freedom-based	0.42	-
English no spaces	Lexicon-based	0.79	-
Russian	Freedom-based	1.0	1.0 (vs 1.0)
Russian	Lexicon-based	0.94	-
Russian no spaces	Freedom-based	0.26	-
Russian no spaces	Lexicon-based	0.72	-
Chinese	Freedom-based	0.71	0.92 (vs 0.94)
Chinese	Lexicon-based	0.83	-

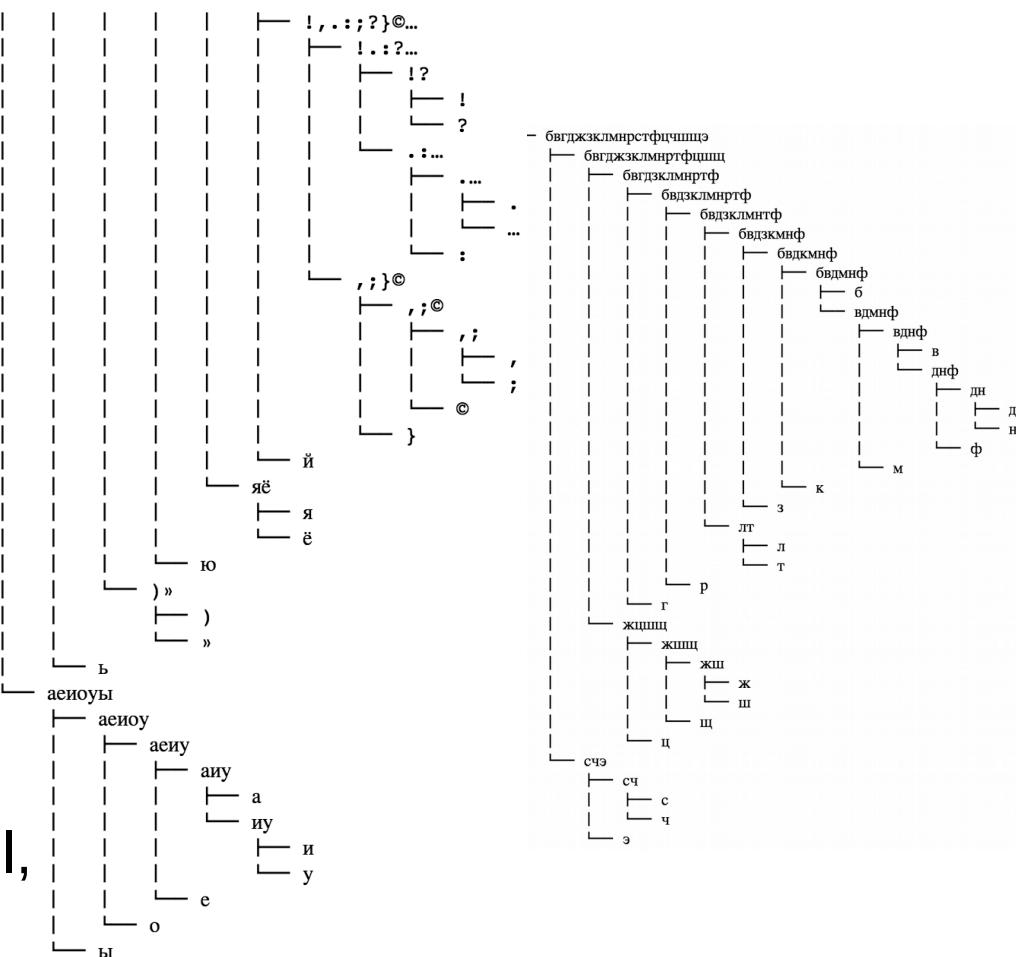
Lexicon-based Tokenization - greedy/beam search on word length (optimal) or frequency

Unsupervised Character Category Learning

Agglomerative Clustering in space of Transitions

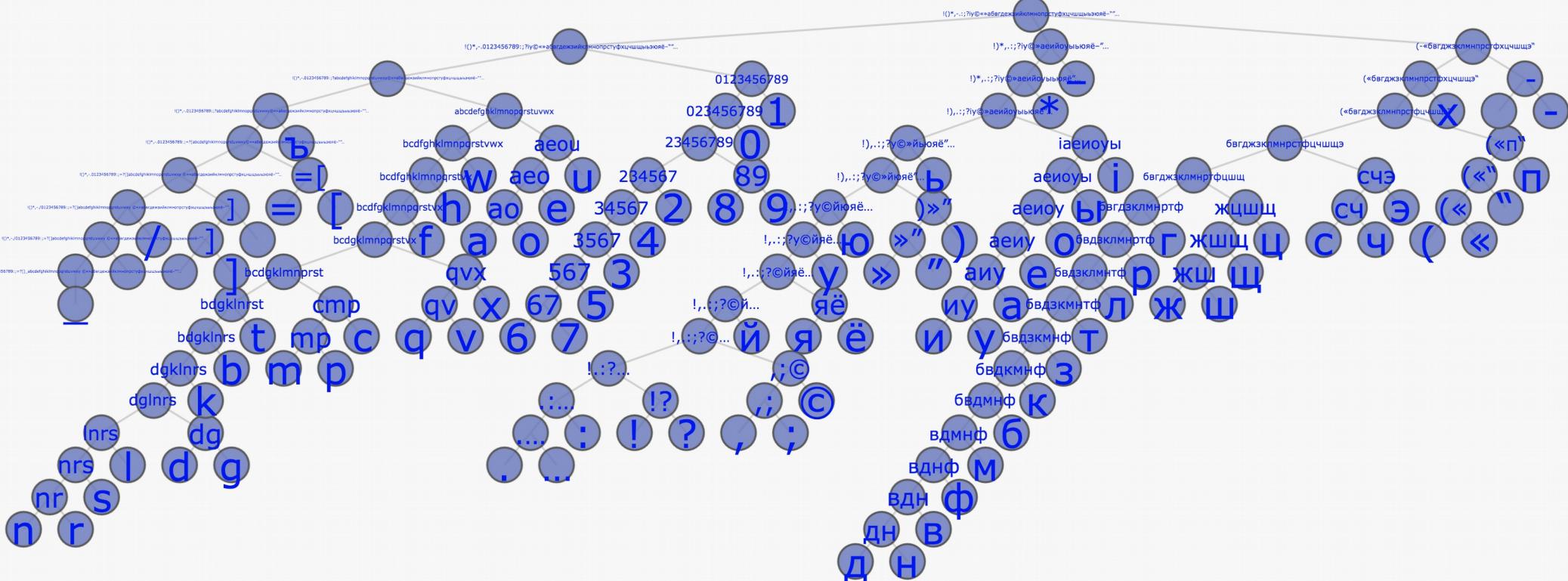


RusAge Test/Small, Cosine Similarity



Unsupervised Character Category Learning

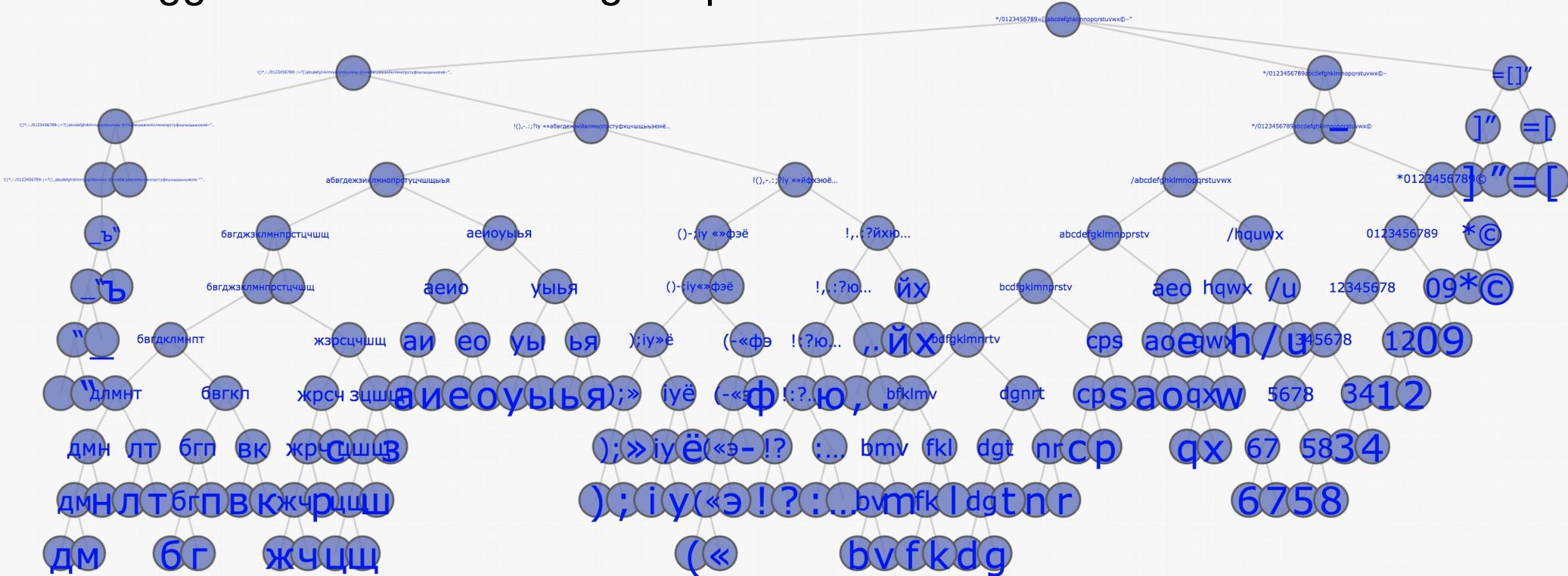
Agglomerative Clustering in space of Transitions



RusAge Previews/Full, Cosine Similarity

Unsupervised Character Category Learning

Agglomerative Clustering in space of Transitions



RusAge Previews/Full, Jackard Similarity

Conclusion and Further Work

Unsupervised Tokenization based on Transition Freedom (TF) recall and precision appears good enough as initial approximation for further applications of self-reinforcement learning as part of interpretable unsupervised learning of natural language.

Optimal thresholds and specific TF-based metrics are specific to language. The process and policy of their discovery and adjustment should be further explored.

Clustering or parts of speech on space of transition graphs may provide some insights on morphology and punctuation structure of low-resource and domain-specific languages.

Hybridization of TF-based tokenization approach with lexicon-based one might be efficient for low-resource and domain-specific languages.

Further unsupervised grammar learning experiments can be run on the basis of suggested unsupervised tokenization approach.

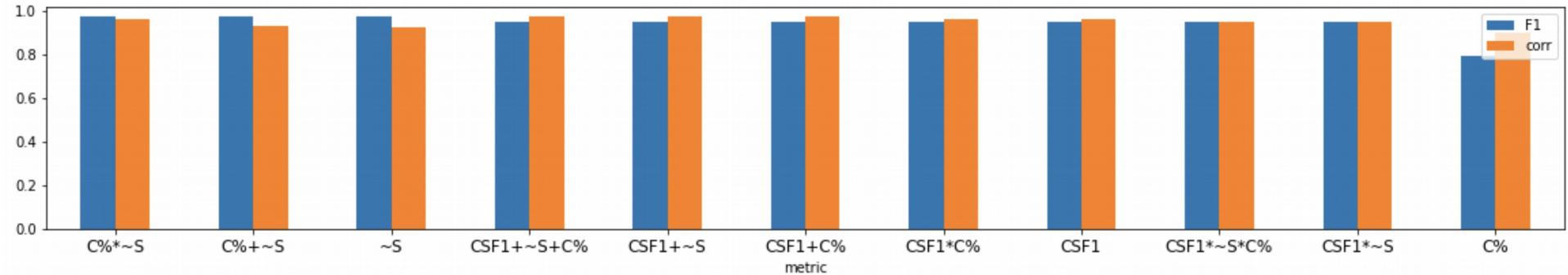
Applications for other Experiential Learning environments, including the ones with delayed/sparse feedback.

Using Reinforcement Learning techniques with self-reinforcement on historical data under Unsupervised Learning setup.

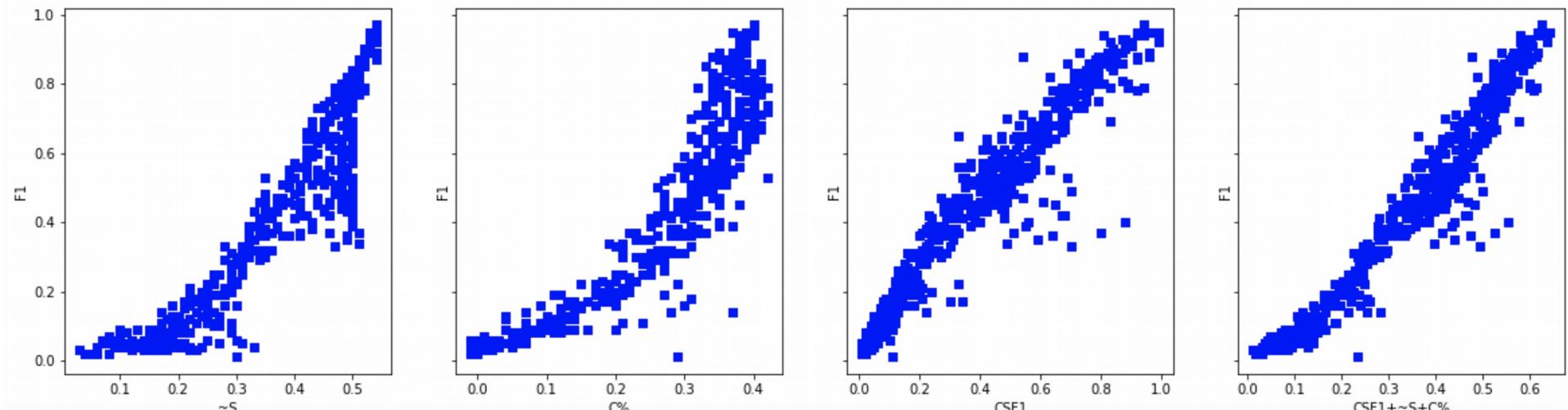
<https://arxiv.org/abs/2205.11443>
<https://github.com/aigents/pygents>

Self-tuning Hyperparameters – English (TF variance)

Test 1000

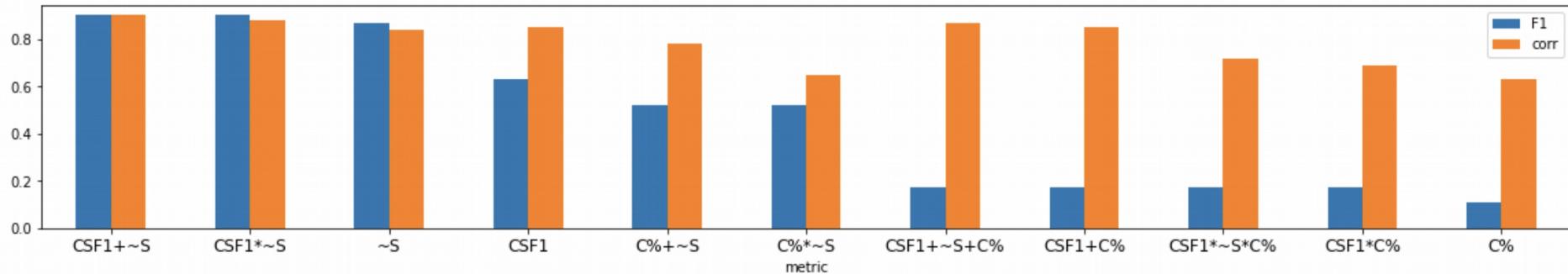


F1 as function of $\sim S$, C% and CSF1 used for hyper-parameter selection

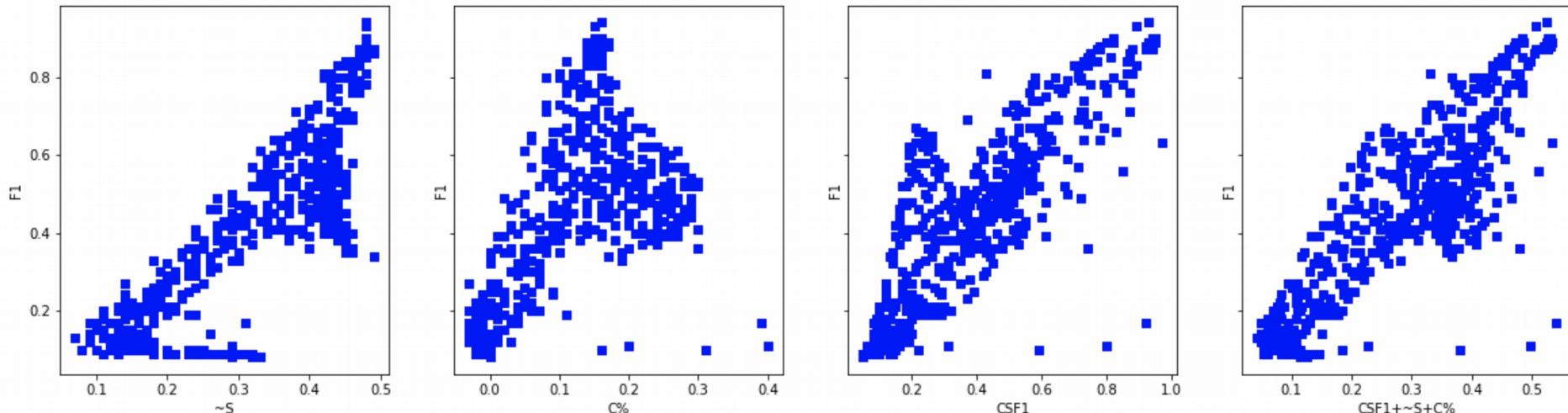


Self-tuning Hyperparameters – Russian (TF variance)

Test 1000

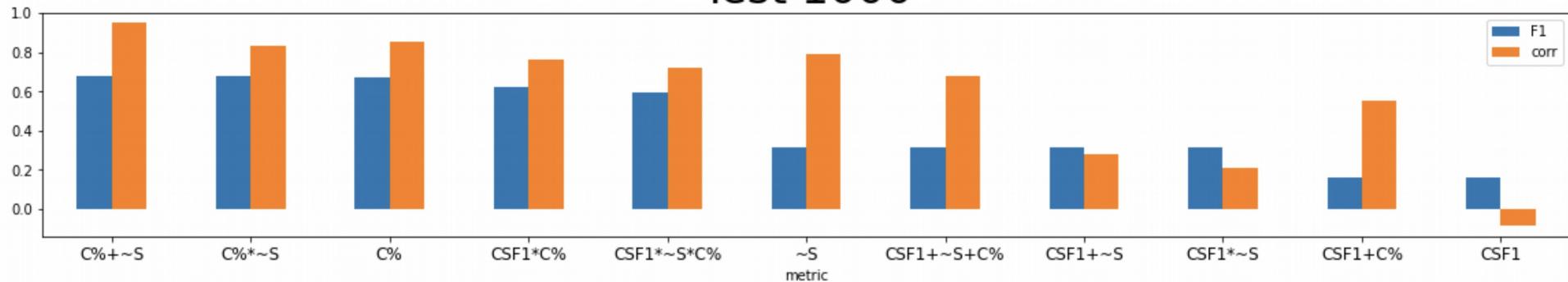


F1 as function of ~S, C% and CSF1 used for hyper-parameter selection

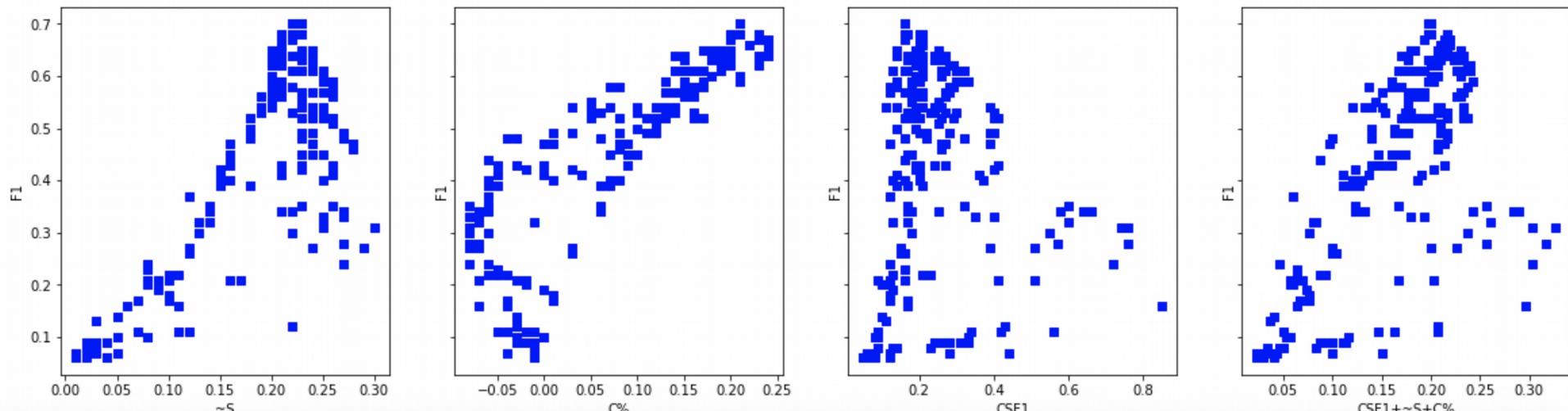


Self-tuning Hyperparameters – Chinese (TF “peak”)

Test 1000



F1 as function of $\sim S$, C% and CSF1 used for hyper-parameter selection



Thank You and Welcome!



<https://agirussia.org>

Anton Kolonin
akolonin@aigents.com
Facebook: akolonin
Telegram: akolonin

