

# Unsupervised Natural Language Text Segmentation and Categorization?

Anton Kolonin

[akolonin@aigents.com](mailto:akolonin@aigents.com)

Facebook: [akolonin](#)

Telegram: [akolonin](#)



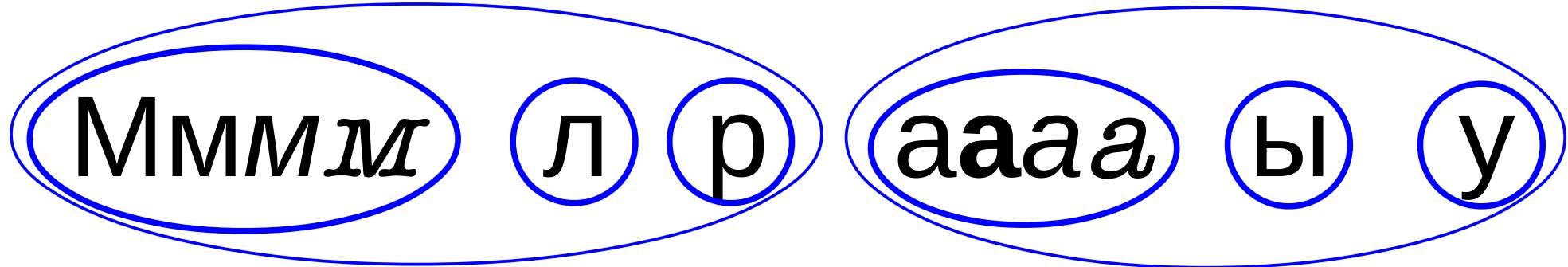
**N\*** Novosibirsk  
State  
University  
**\*THE REAL SCIENCE**

<https://www.nsu.ru/>

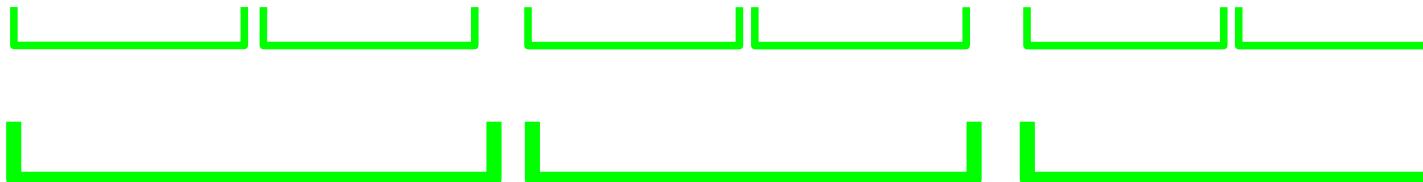


<https://agirussia.org>

# Clustering and Segmentation



Мама мыла раму



# Clustering and Segmentation

— м а л о ы н ш у

ма ло мы ла на шу

мало\_мама\_мыла\_Нашу\_Машу\_мылом

# Clustering and Segmentation

— м а л о ы н ш у

ма|ло ма|ма мы|ла на|шу ма|шу мы|лом

мало\_|\_мама\_|\_мыла\_|\_нашу\_|\_машу\_|\_мылом

мало\_мама\_мыла\_нашу\_машу\_мылом

# Clustering and Segmentation

— м а л о ы н ш у

м|а л|о м|ы л|а н|а ш|у

ма ло мы ла на шу м

ма|ло ма|ма мы|ла на|шу ма|шу мы|лом

мало мама мыла нашу машу мылом

мало\_|\_мама\_|\_мыла\_|\_нашу\_|\_машу\_|\_мылом

мало\_мама\_мыла\_нашу\_машу\_мылом

# Clustering and Segmentation

мама\_мыла\_машу маша\_мыла\_маму

маша\_ела\_кашу мама\_ела\_кашу

маша\_ела\_суши мама\_ела\_суши

маша\_ела\_мало\_каши

мама\_ела\_мало\_суши

# 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>

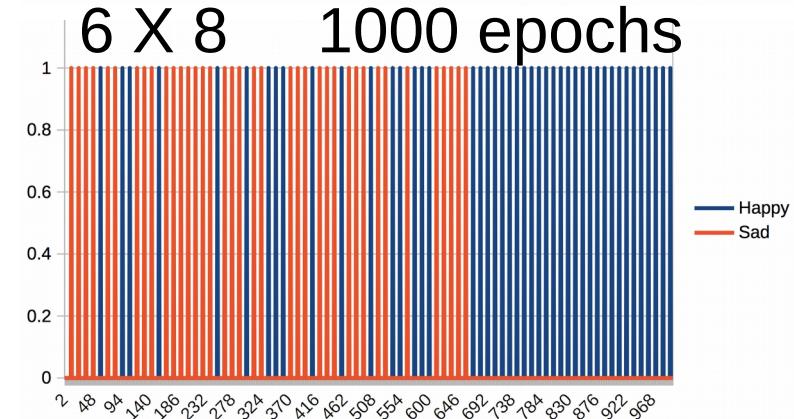
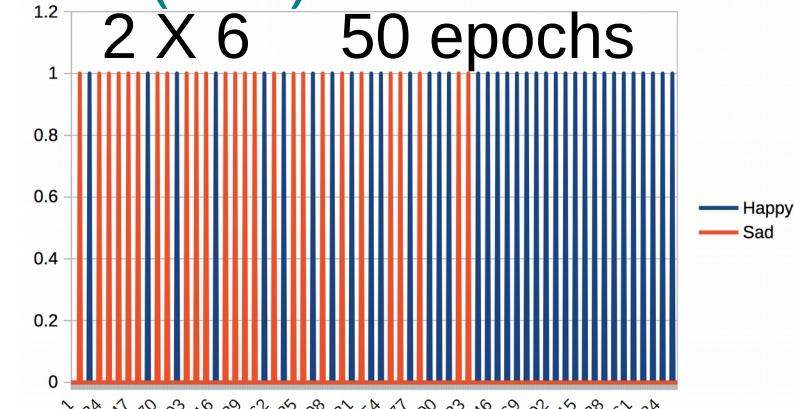
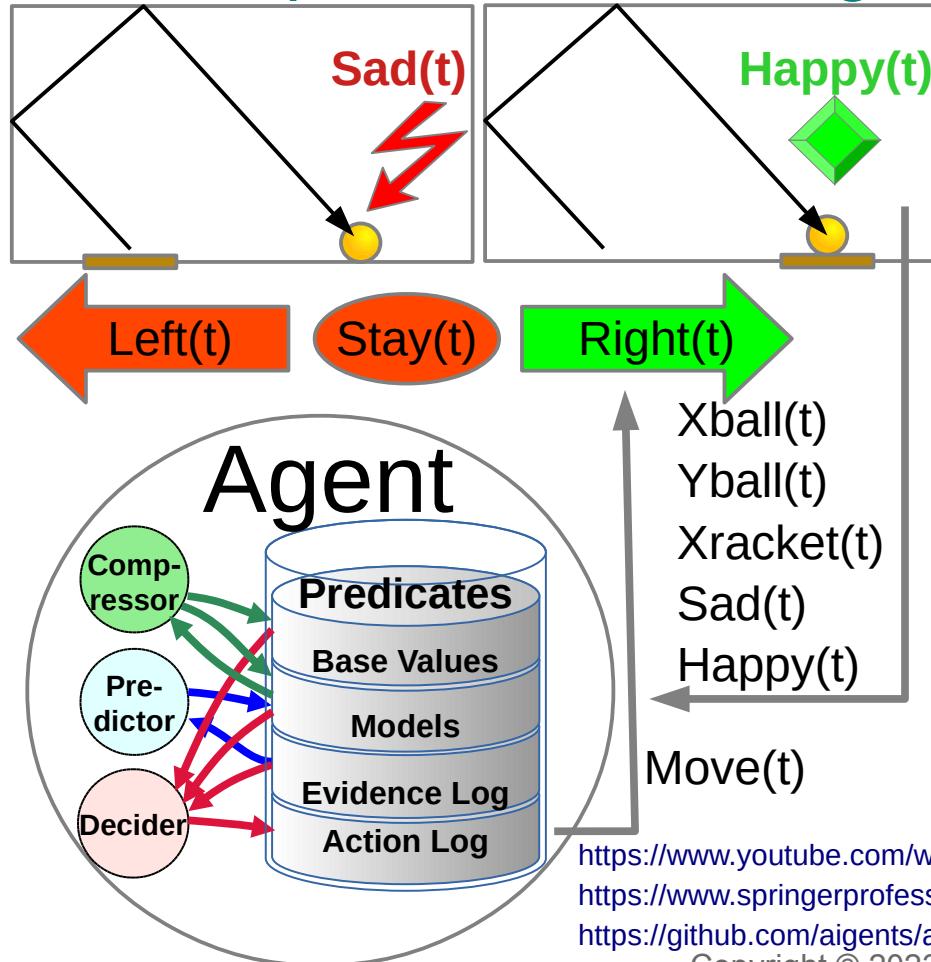
<https://github.com/aigents/pygents>

**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

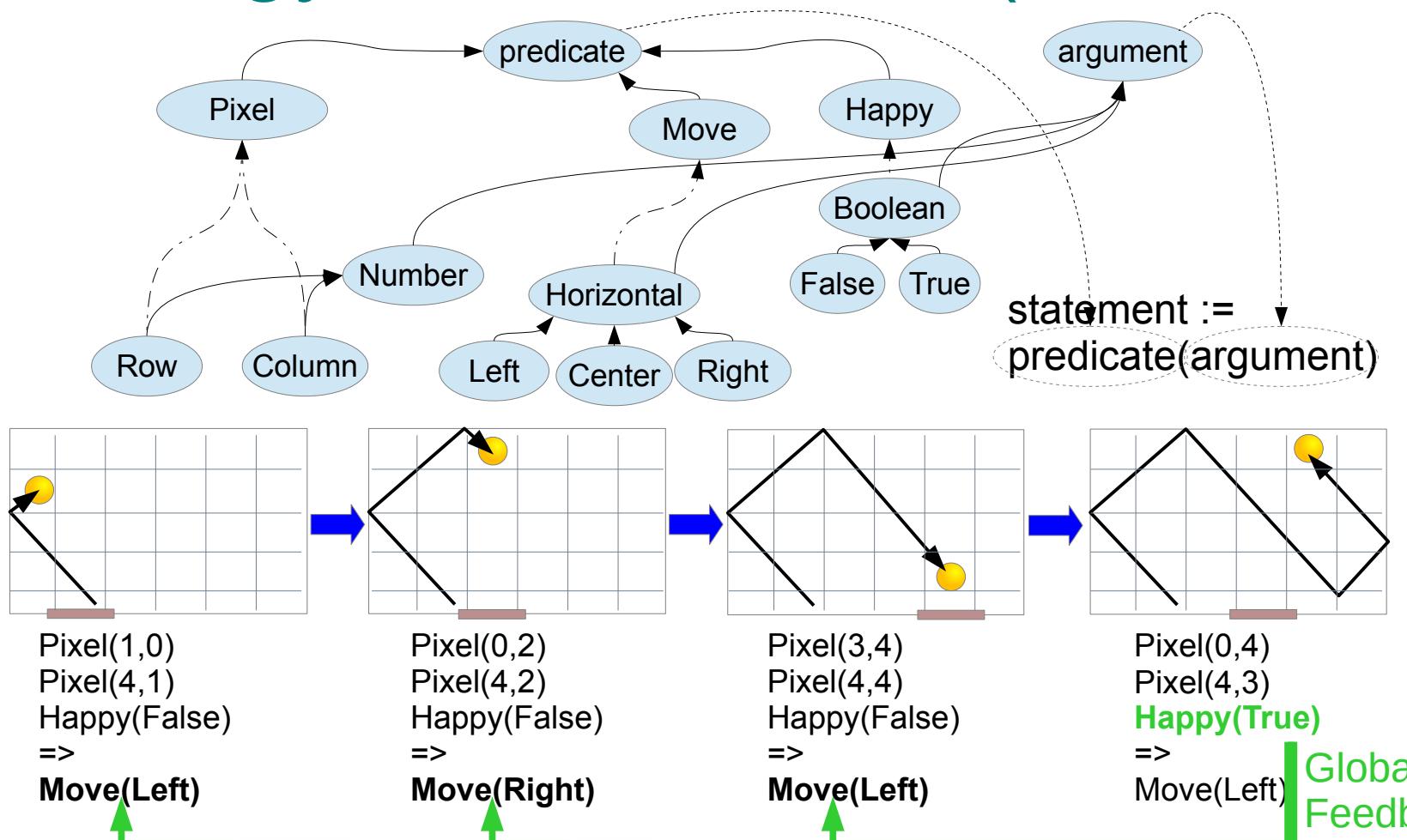


<https://www.youtube.com/watch?v=2LPLhJKh95g>

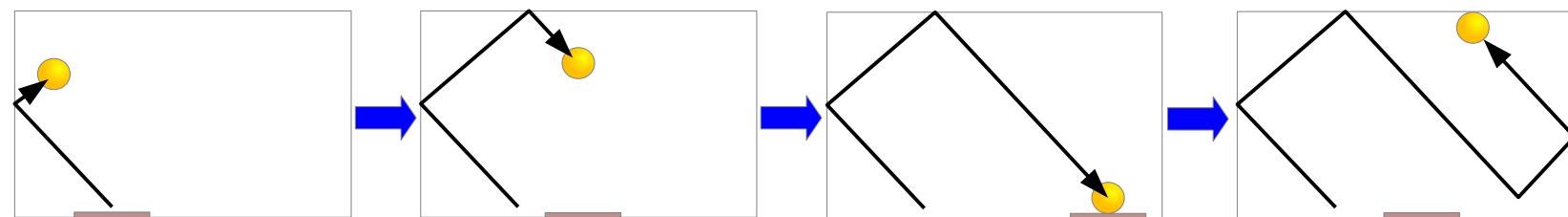
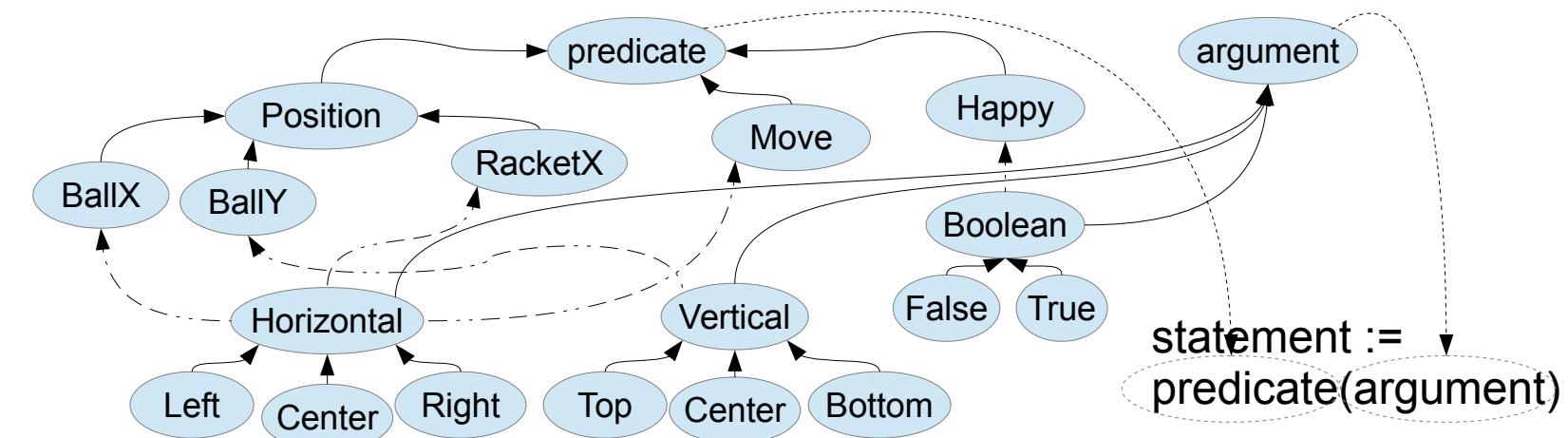
<https://www.springerprofessional.de/neuro-symbolic-architecture-for-experiential-learning-in-discret/20008336>

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

# Ontology and Grammar (“Discrete”)



# Ontology and Grammar (“Symbolic”)



BallY(Top)  
BallX(Left)  
RacketX(Left)  
Happy(False)  
=> **Move(Left)**

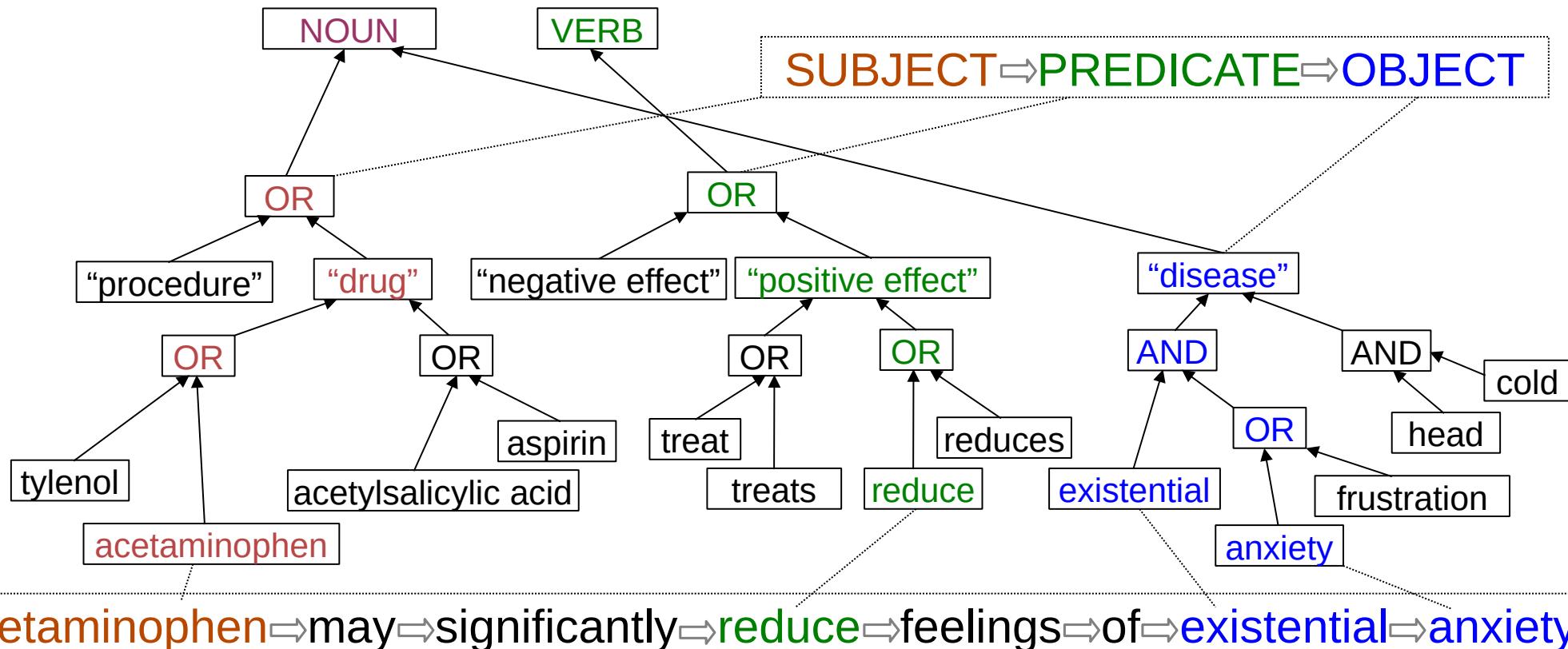
BallY(Top)  
BallX(Center)  
RacketX(Center)  
Happy(False)  
=> **Move(Right)**

BallY(Bottom)  
BallX(Right)  
RacketX(Right)  
Happy(False)  
=> **Move(Right)**

BallY(Bottom)  
BallX(Right)  
RacketX(Right)  
**Happy(True)**  
=> **Move(Left)**

Global  
Feedback

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/>

# Unsupervised Segmentation

No stop/start/break tokens, lexicons and  
rules are known!!!

# 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

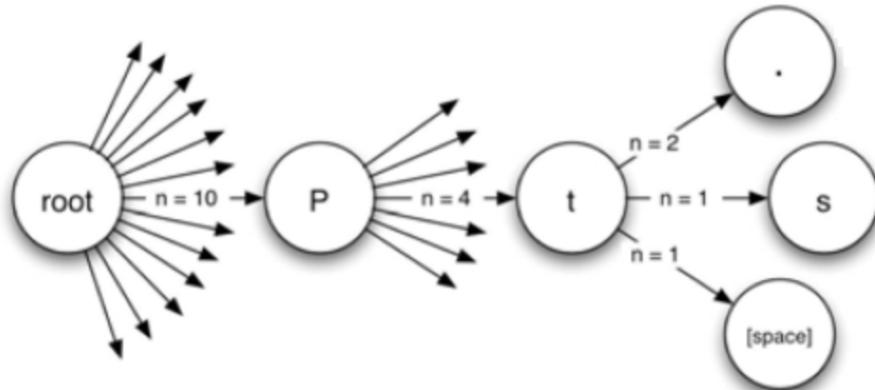
<https://www.springerprofessional.de/unsupervised-language-learning-in-opencog/15995030>

<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



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.

Copyright ©2007 AMIA - All rights reserved. Jesse O. Wrenn, Peter D. Stetson, and Stephen B. Johnson. 2007. An unsupervised machine learning approach to segmentation of clinician-entered free text. AMIA Annu Symp Proc. 2007; 2007: 811–815.

## Metrics/Indicators:

### Mutual Information<sup>1</sup>

### Conditional Probability<sup>1,2</sup>

### Transition Freedom<sup>2,3</sup>

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

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

<sup>3</sup> 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) →  
P(The roses in the vase by the door is) →

Is the correct answer ranked higher?  
 $P(\dots\text{are}) > P(\dots\text{is})$

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.

is/are	
The roses <u>?</u>	Simple: no attractors
The roses in the <u>vase</u> <u>?</u>	Harder: 1 attractor
The roses in the <u>vase</u> by the <u>door</u> <u>?</u>	Harder: 2 attractors

Attractors: nouns with different number than the subject

[https://lena-voita.github.io/nlp\\_course/language\\_modeling.html](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 ( $dvp-/dvp+$ ) from mean

Derivative ( $dp-/dp+$ ) and “Peak”

### Transition Freedom ( $f-/f+$ )

Deviation ( $dvf-/dvf+$ ) from mean

Derivative ( $df-/df+$ ) and “Peak”

## Hyper-parameters:

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

Threshold for model compression

Threshold for segmentation

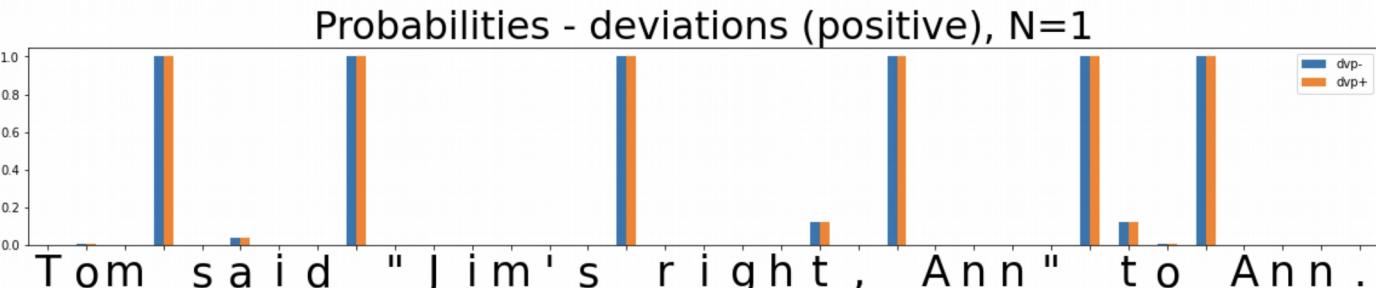
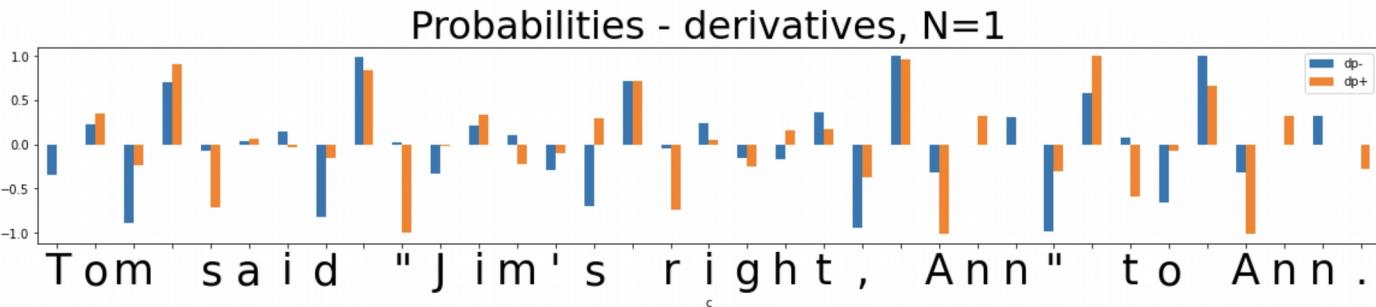
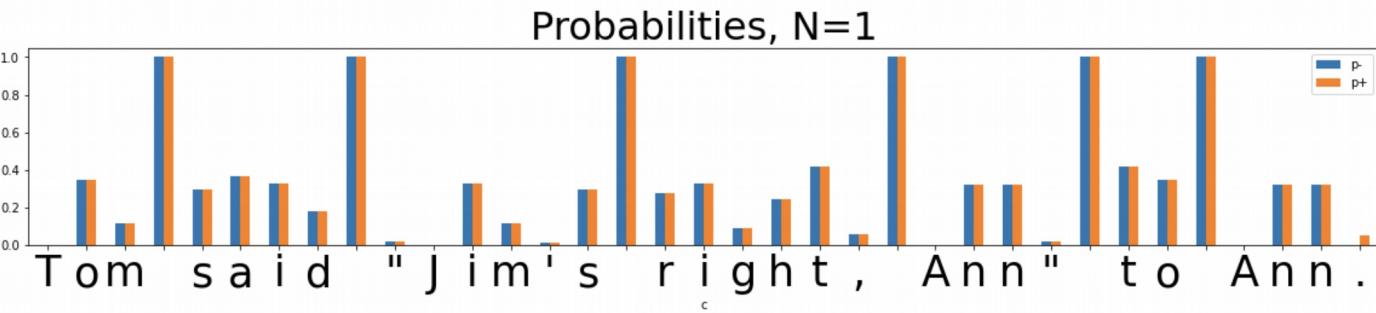
## Evaluations:

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



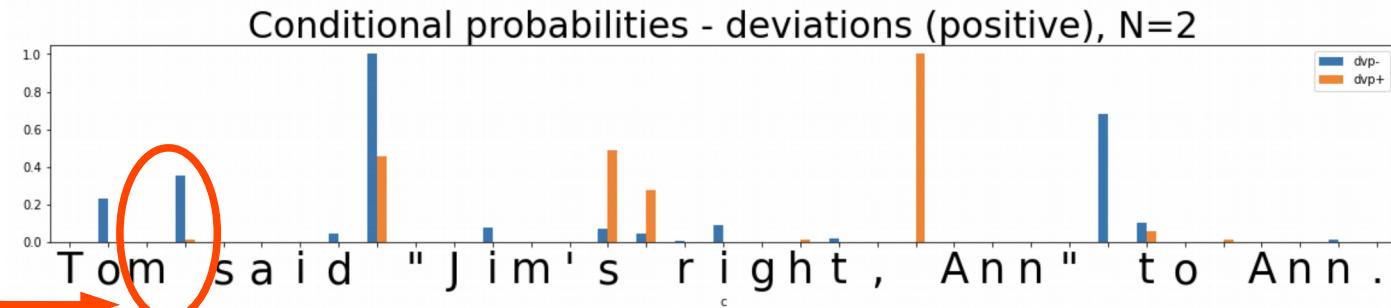
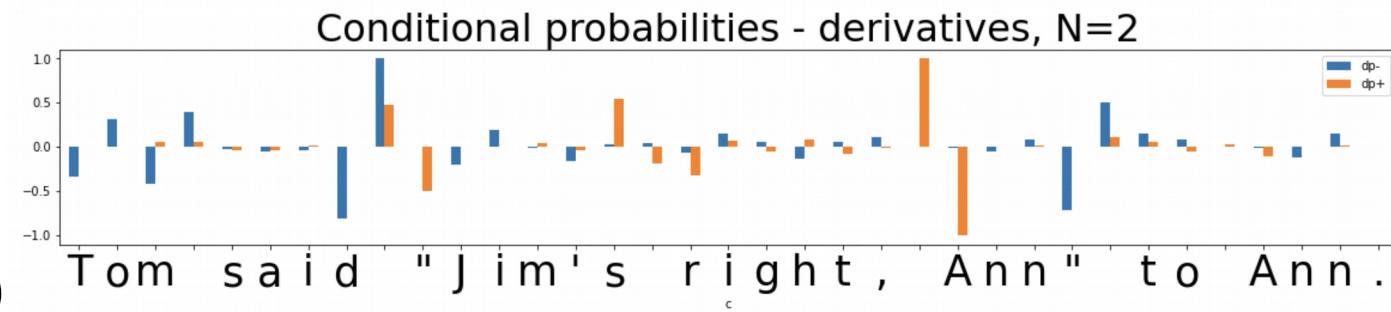
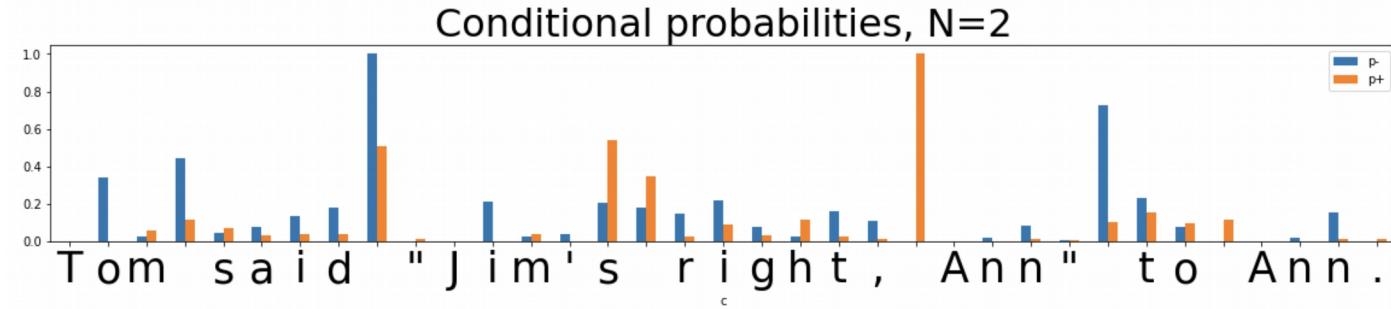
# Unsupervised Text Segmentation (Tokenization)

## Metrics/Indicators:

Ngram (Character)  
Conditional  
Probability  
(of Transition)

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

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



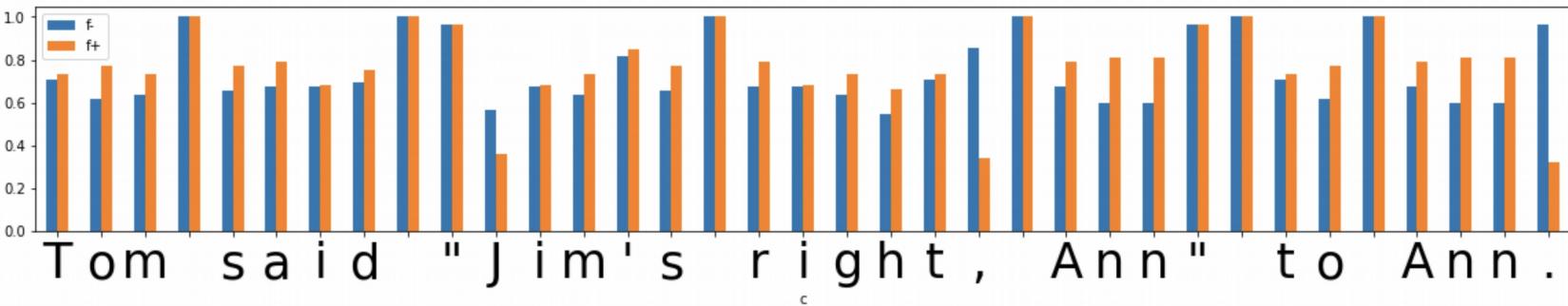
# Unsupervised Text Segmentation (Tokenization)

```
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
```

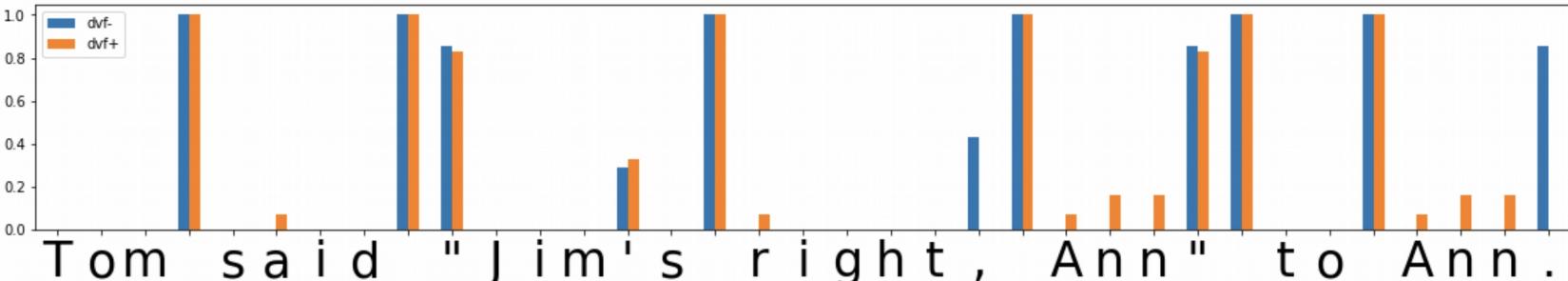
```
Threshold 0.35
Tom said "Jim's right, Ann" to Ann.
['Tom', ' ', 'said', ' ', "'", "Jim's", ' ', 'right', ' ', ' ', 'Ann', ' ', ' ', 'to', ' ', ' ', 'Ann', '.']
['Tom', ' ', 'said', ' ', "'", "Jim's", ' ', 'right', ' ', ' ', 'Ann', ' ', ' ', 'to', ' ', ' ', 'Ann', '.']
1.0
```

## Metrics/ Indicators:

Transition  
Freedom  
(Freedom of  
Transition)



Transition  
Freedom  
Deviation



# Unsupervised Text Segmentation (Tokenization)

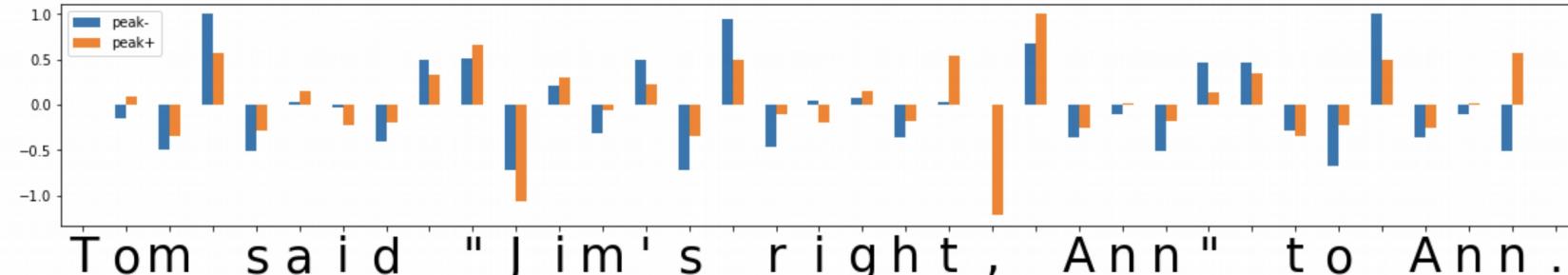
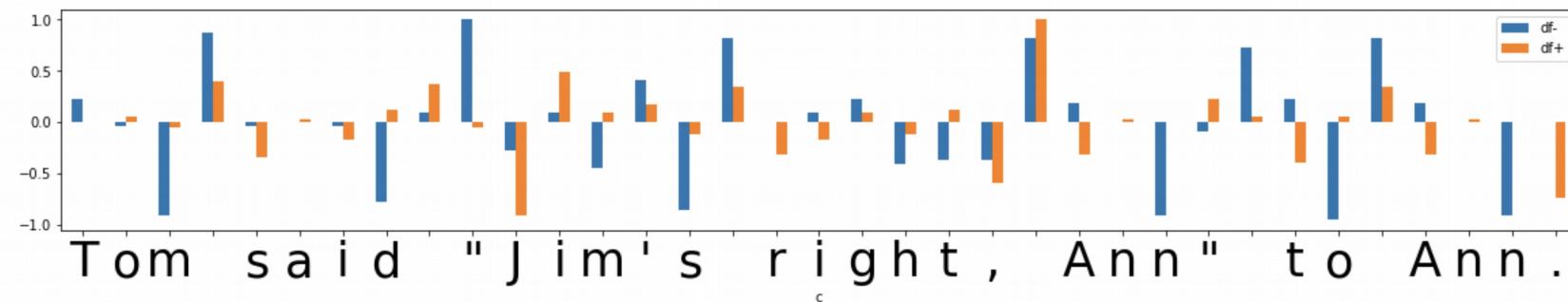
## Metrics/ Indicators:

Transition  
Freedom  
(Freedom of  
Transition)  
Derivative

Transition  
Freedom  
“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

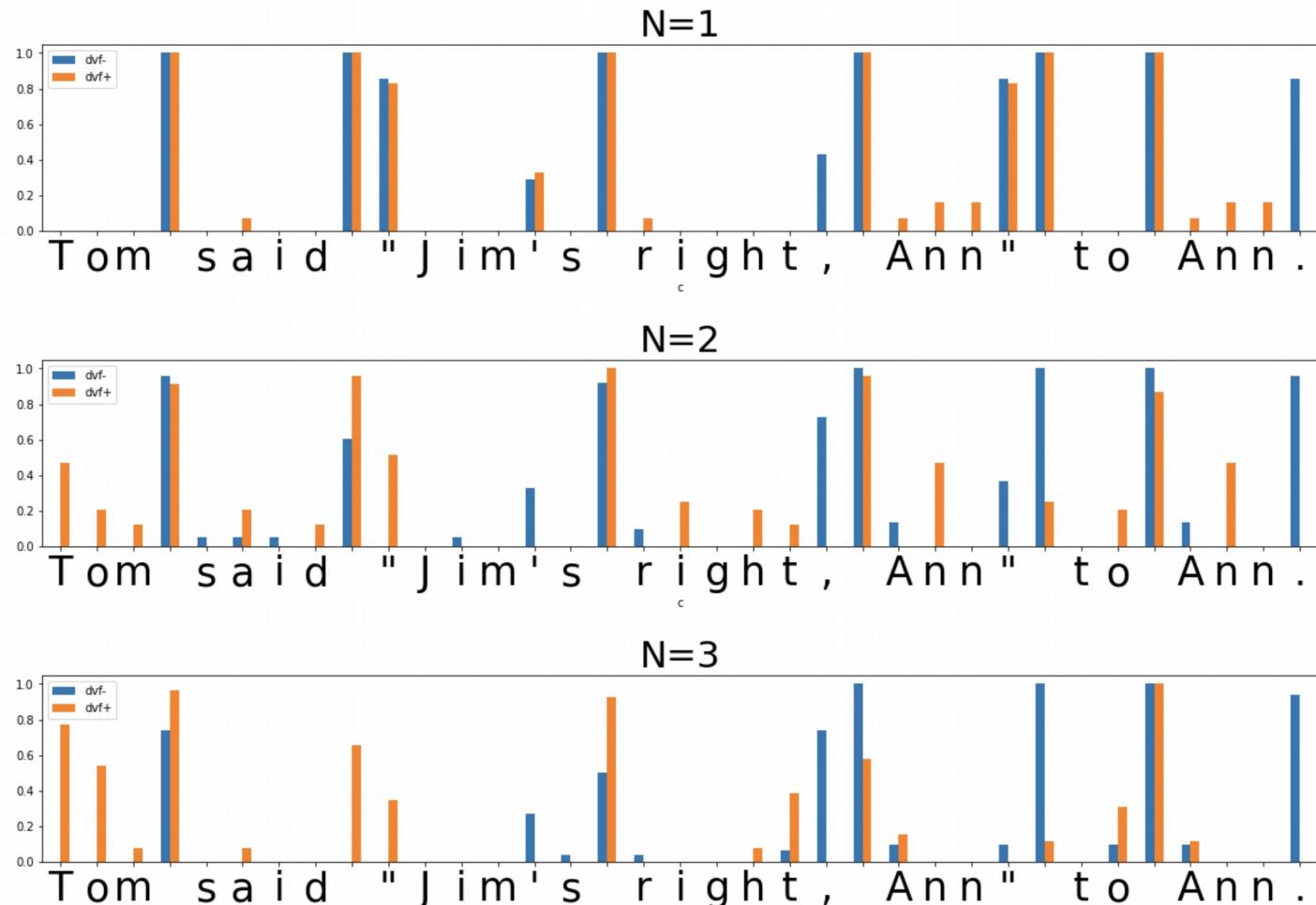


# Unsupervised Text Segmentation (Tokenization)

Metrics/  
Indicators:

Transition  
Freedom  
Deviation

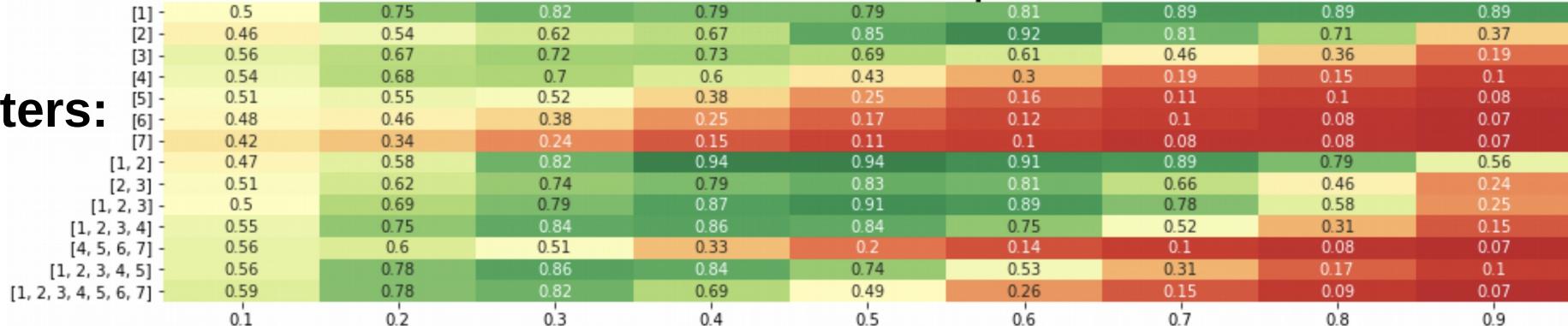
(varying “N”  
of N-gram)



# Unsupervised Text Segmentation (Tokenization)

## English

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



## Hyper-Parameters:

Metric:  
Transition  
Freedom

Threshold  
for model  
compression

Combination  
of Ngram N-s

Threshold for  
segmentation

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



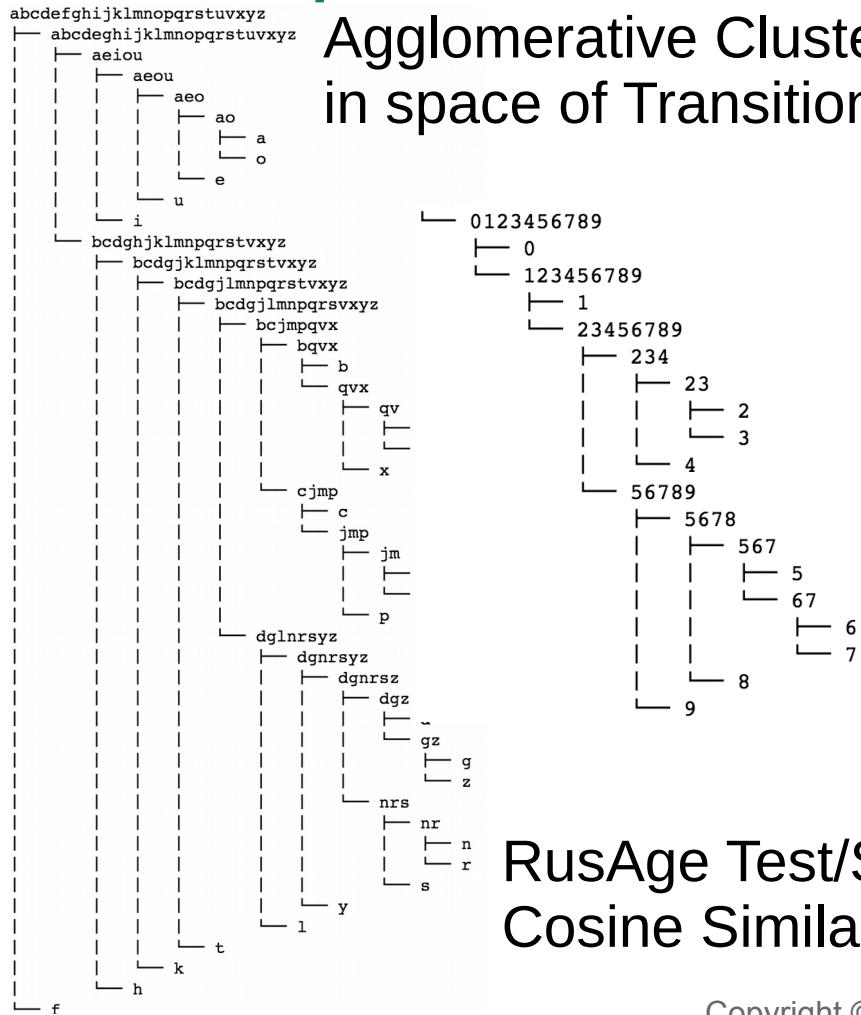
# Results – Freedom-based Tokenization against Lexicon

Language	Tokenizer	Tokenization F1	Lexicon Discovery Precision
English	Freedom-based	<b>0.99</b>	<b>0.99</b> (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	<b>1.0</b>	<b>1.0</b> (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	<b>0.71</b>	<b>0.92</b> (vs 0.94)
Chinese	Lexicon-based	0.83	-

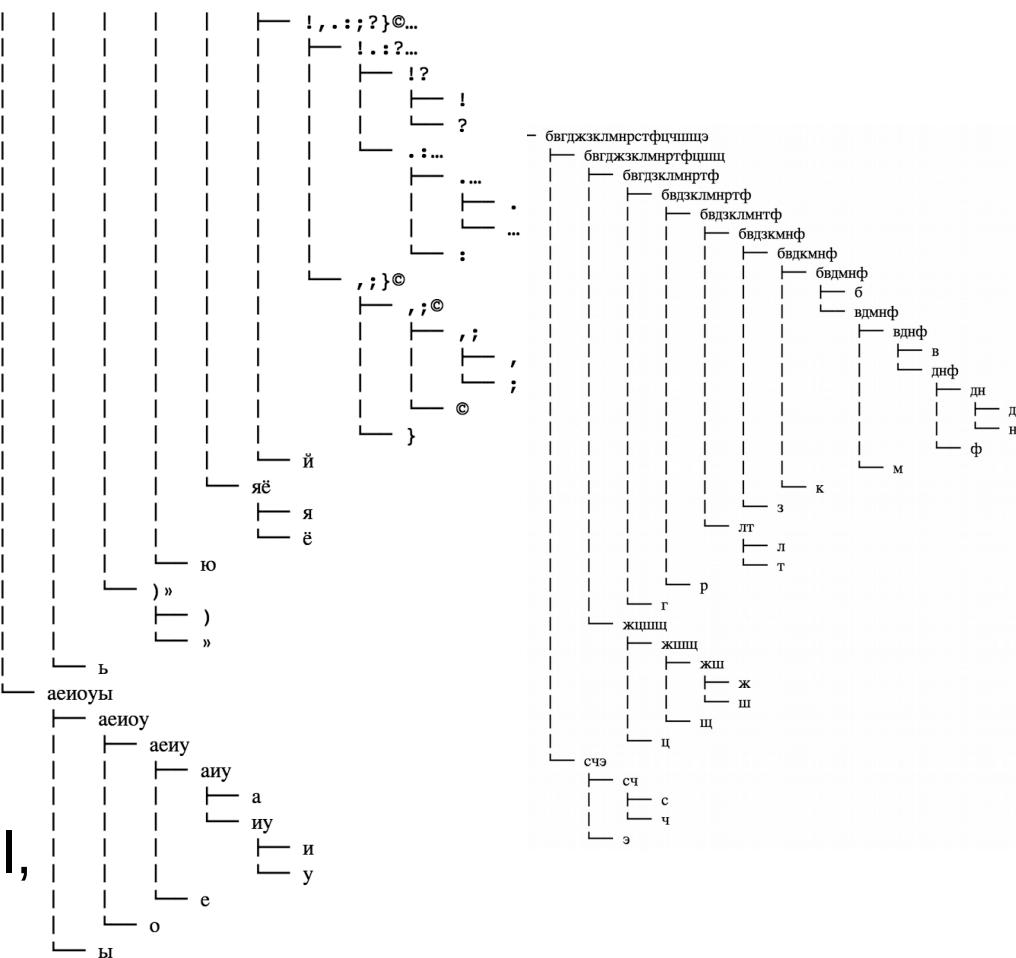
*Lexicon-based Tokenization - greedy/beam search on word length (optimal) or frequency*

# Unsupervised Character Category Learning

# Agglomerative Clustering in space of Transitions



# RusAge Test/Small, Cosine Similarity



# 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.

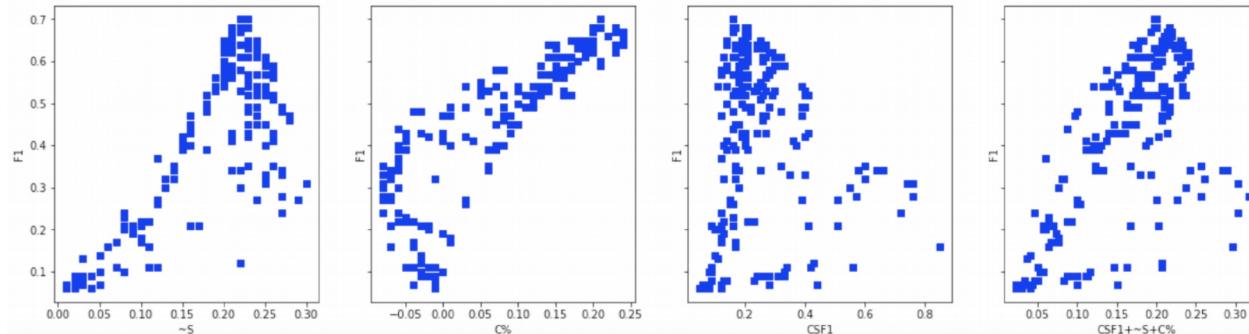
Applications for other Experiential Learning environments, including the ones with delayed/sparse feedback.

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

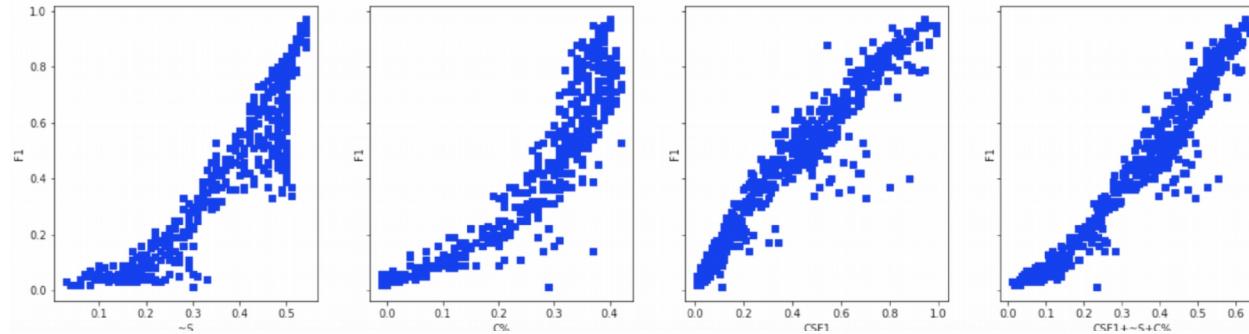
<https://arxiv.org/abs/2205.11443>  
<https://github.com/aigents/pygents>

# Which language is English, Chinese, Russian?

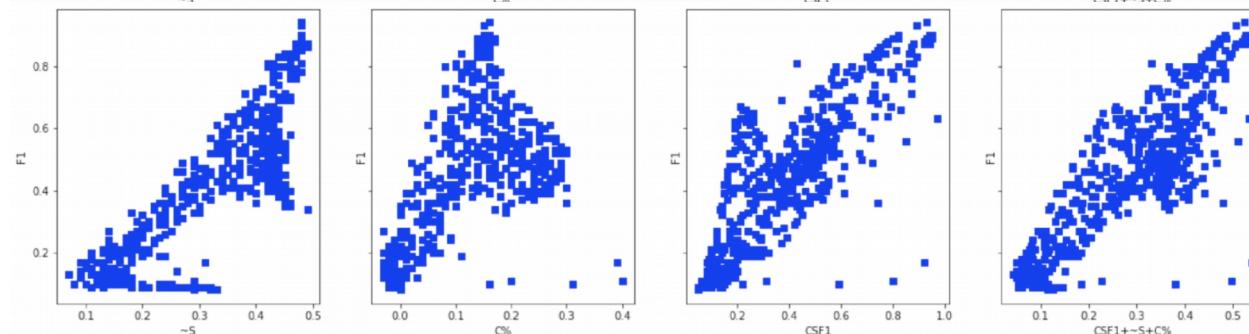
Language 1



Language 2

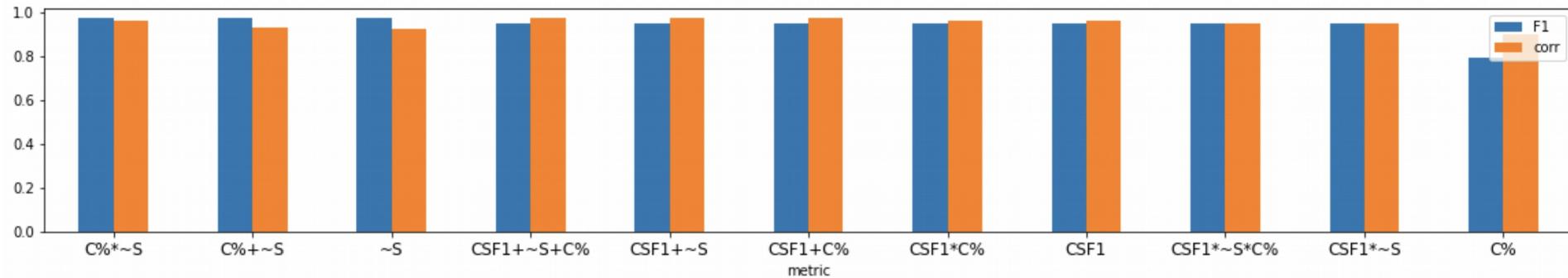


Language 3

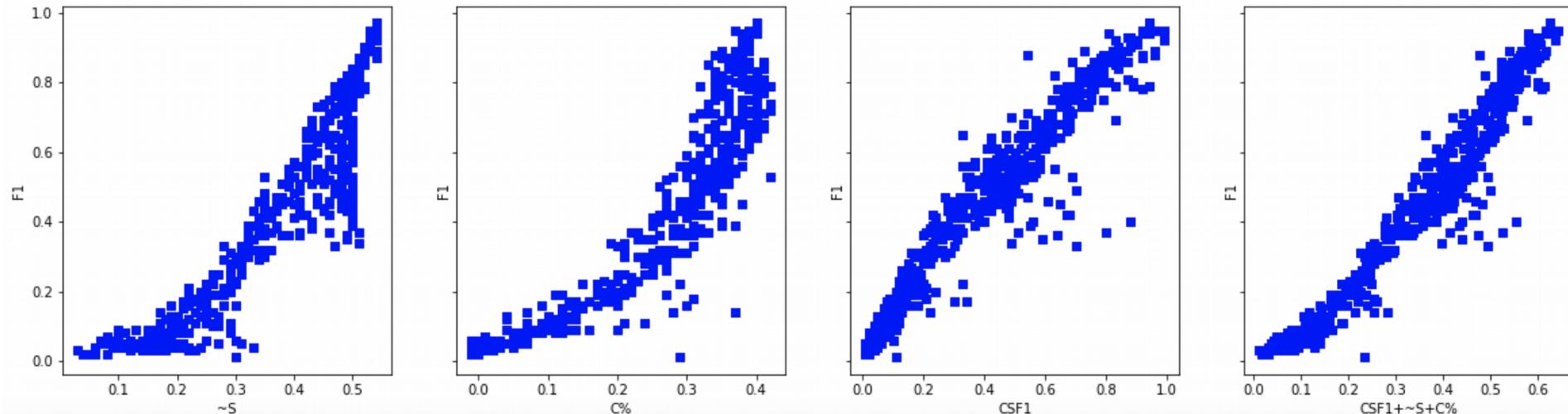


# Self-tuning Hyperparameters – English (TF variance)

Test 1000



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



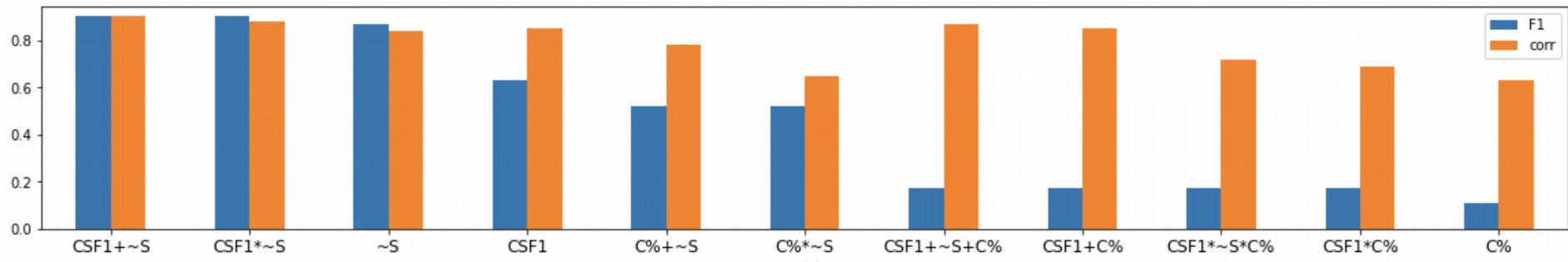
[https://github.com/agents/pygents/blob/main/notebooks/nlp/english/en\\_tokenizer\\_auto.ipynb](https://github.com/agents/pygents/blob/main/notebooks/nlp/english/en_tokenizer_auto.ipynb)

Copyright © 2023 Anton Kolonin, Aigents®

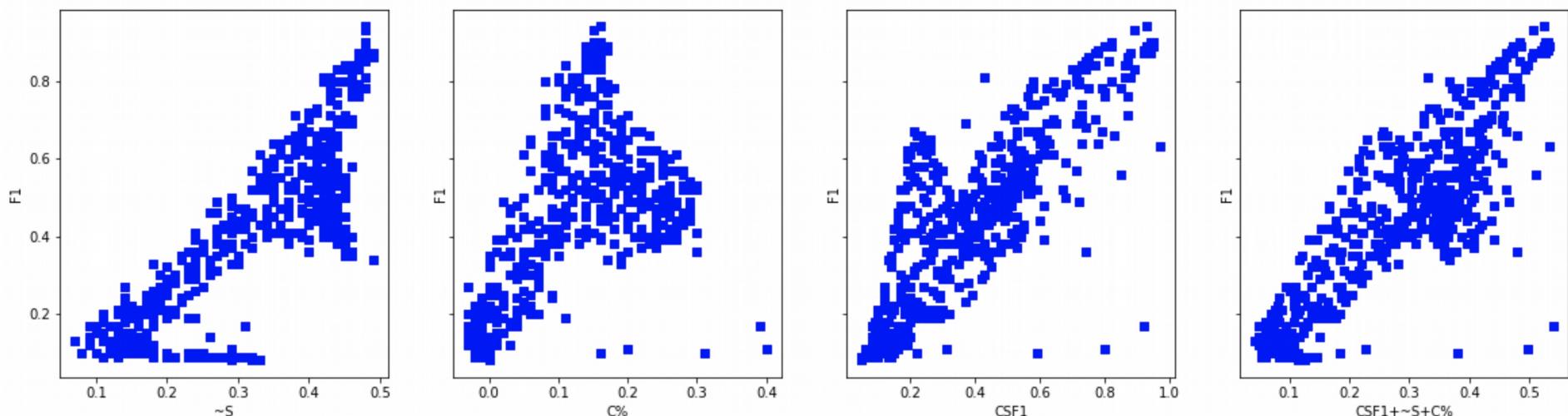
# Self-tuning Hyperparameters – Russian (TF variance)



Test 1000



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



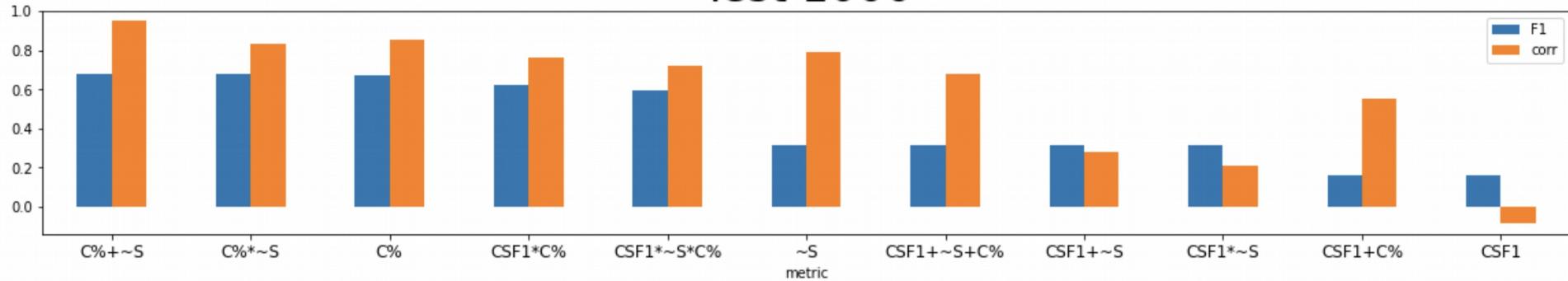
[https://github.com/aigents/pygents/blob/main/notebooks/nlp/russian/ru\\_tokenizer\\_auto.ipynb](https://github.com/aigents/pygents/blob/main/notebooks/nlp/russian/ru_tokenizer_auto.ipynb)

Copyright © 2023 Anton Kolonin, Aigents®

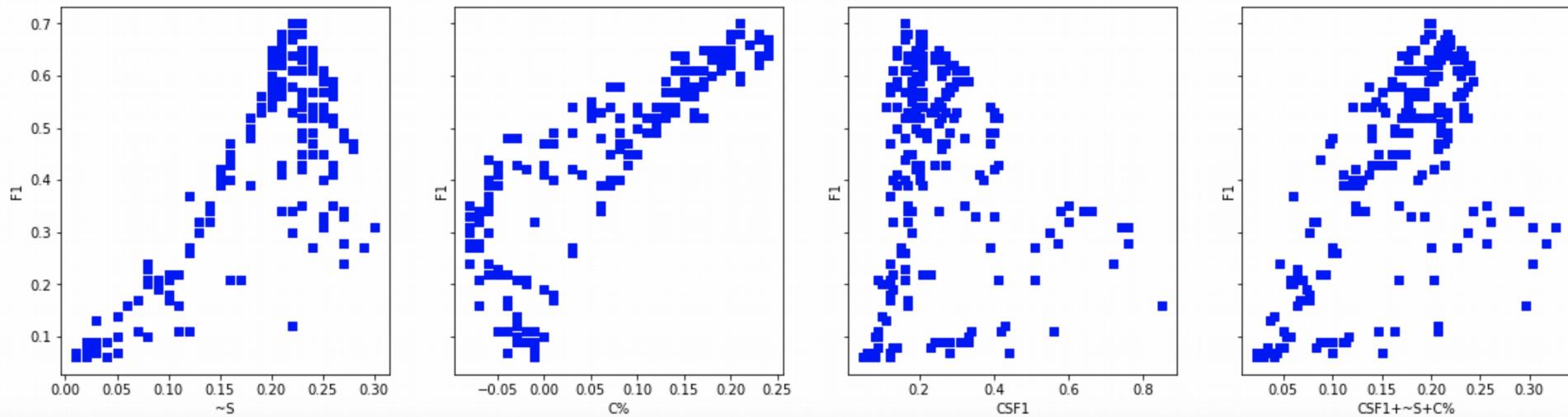
# Self-tuning Hyperparameters – Chinese (TF “peak”)



Test 1000



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



# Something about Human Intuition!

Screen Shot 2022-06-16 at 11.08.54.png  
247.8 KB

OPEN WITH

Language 1  11:22 ✓

Screen Shot 2022-06-16 at 11.09.45.png  
256.8 KB

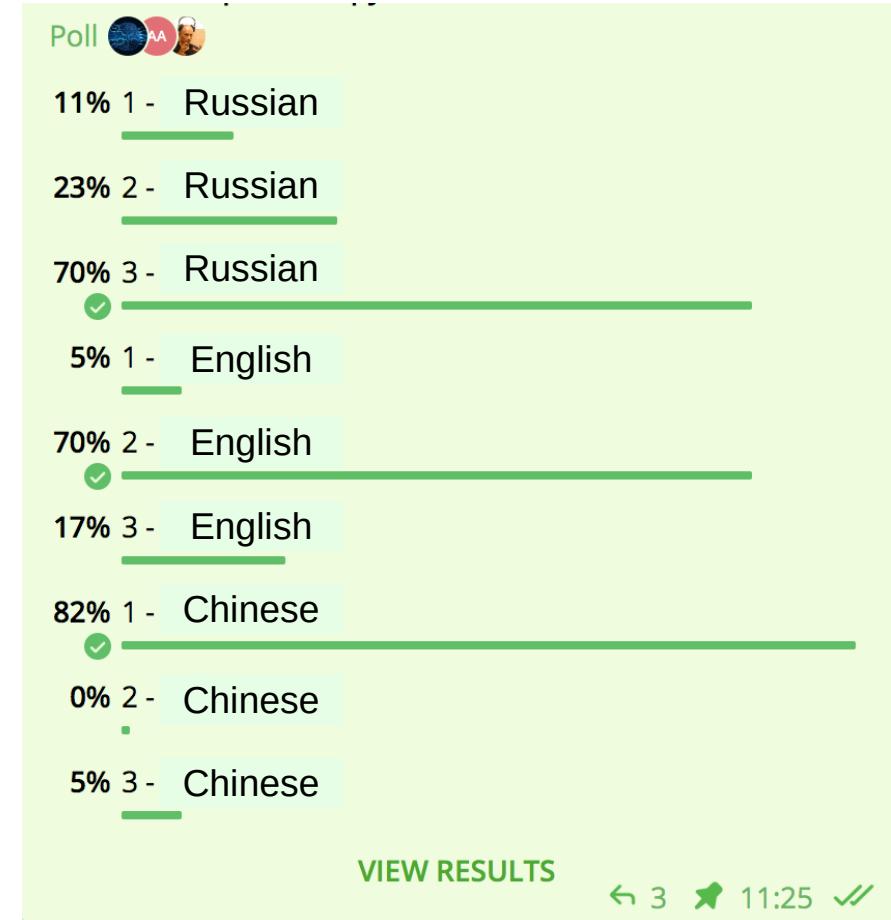
OPEN WITH

Language 2  11:23 ✓

Screen Shot 2022-06-16 at 11.09.59.png  
276.4 KB

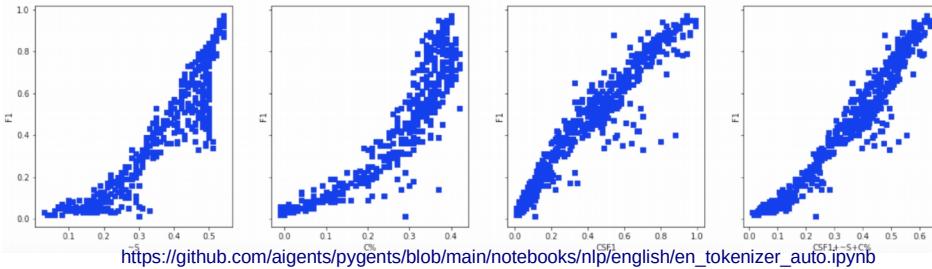
OPEN WITH

Language 3  11:23 ✓

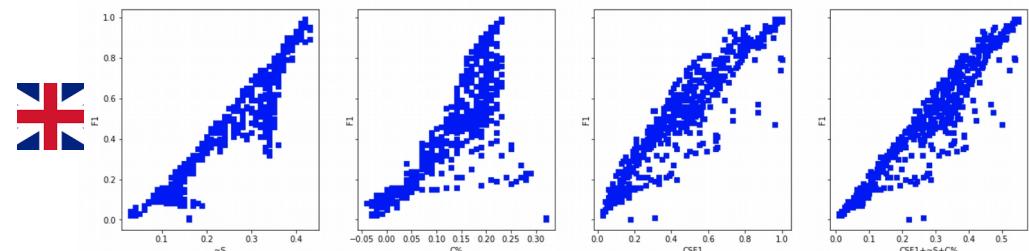


# Different corpora, different sizes

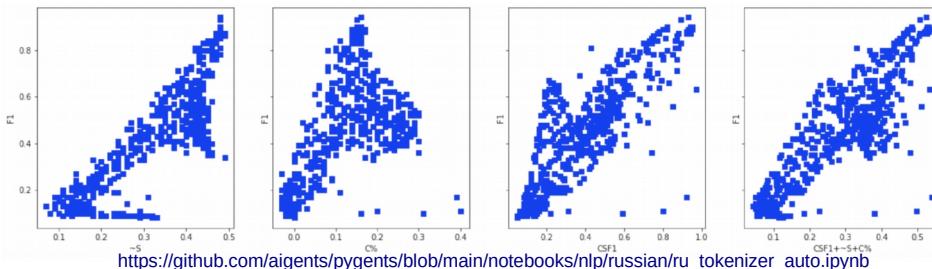
English, Train: Brown, Test: Brown 1000



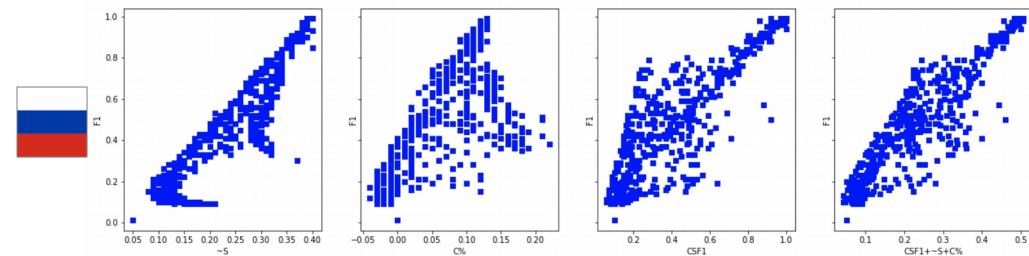
English, Train: Brown, Test: MagicData 100



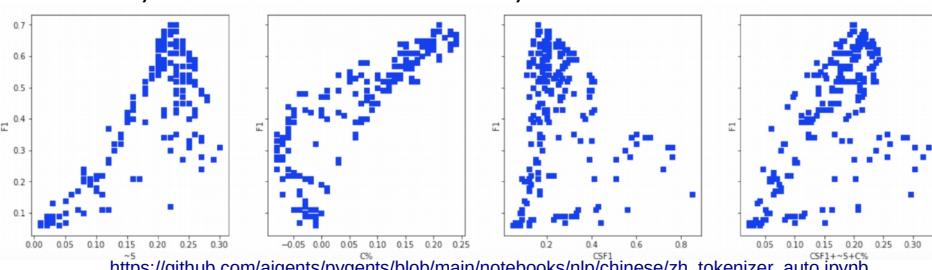
Russian, Train: RusAge, Test: RusAge 1000



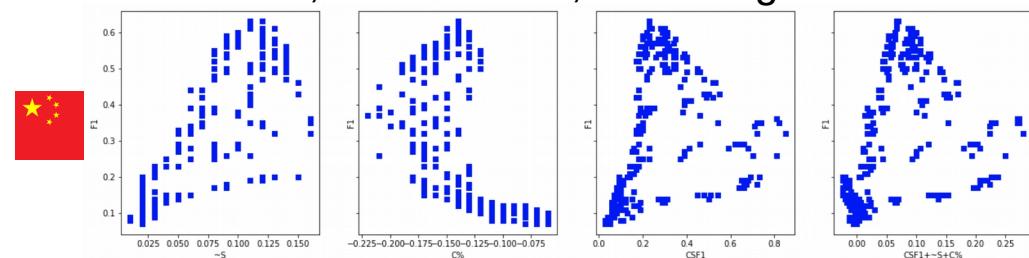
Russian, Train: Brown, Test: MagicData 100



Chinese, Train: CLUE News, Test: CLUE News 1000



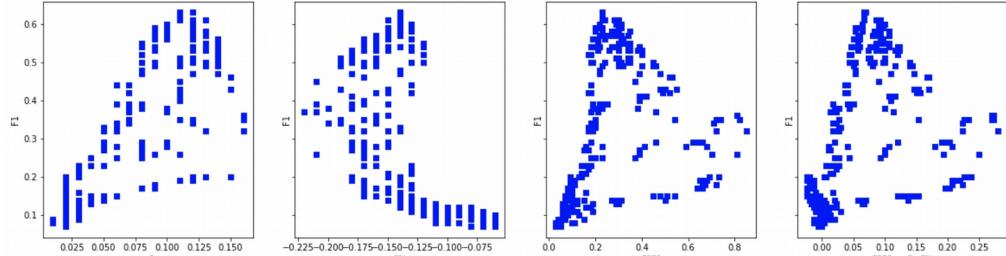
Chinese, Train: Brown, Test: MagicData 100



# Chinese corpora, different sizes

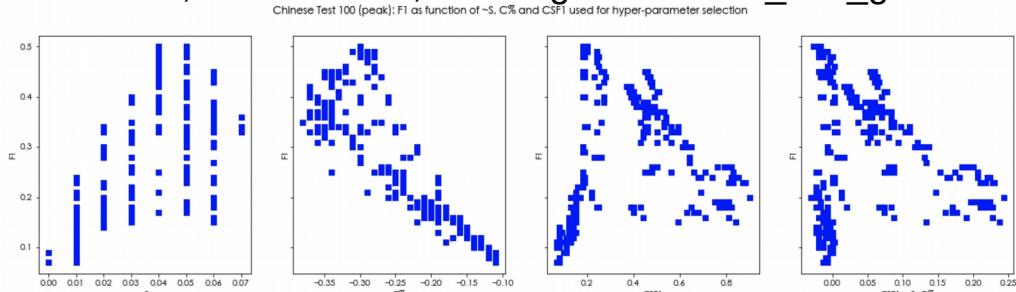


Chinese, Train: Brown, Test: MagicData 100



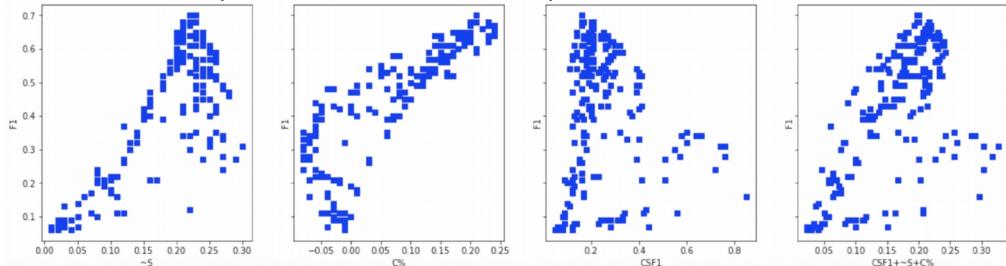
[https://github.com/agents/pygents/blob/main/notebooks/nlp/tokenization/brown/tokenization\\_brown\\_en\\_ru\\_zh.ipynb](https://github.com/agents/pygents/blob/main/notebooks/nlp/tokenization/brown/tokenization_brown_en_ru_zh.ipynb)

Chinese, Train: Brown, Test: sighan2005/as\_test\_gold 100

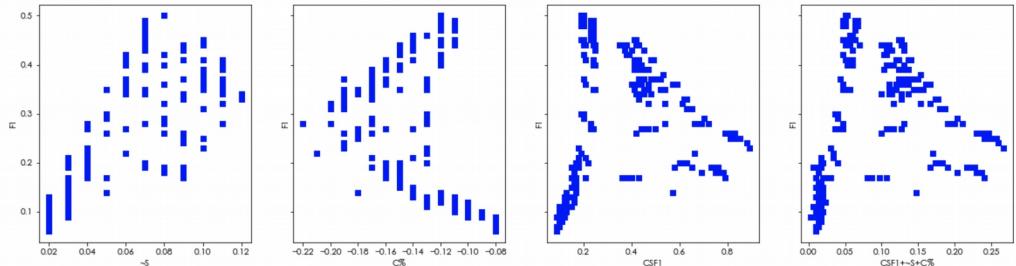


Chinese Test 100 (peak): F1 as function of -S, C% and CSF1 used for hyper-parameter selection

Chinese, Train: CLUE News, Test: CLUE News 1000

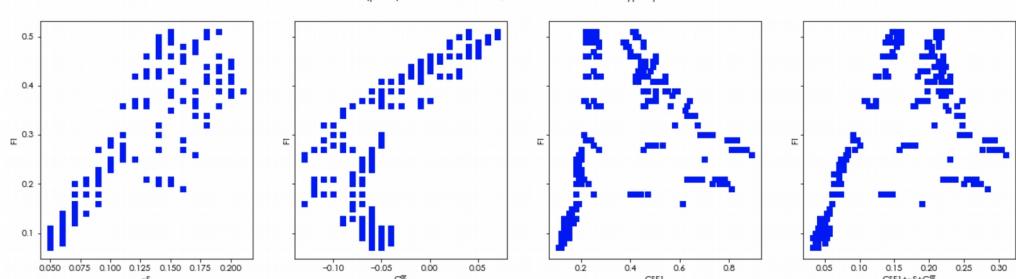


Chinese, Train: CLUE News, Test: CLUE News 10000



Chinese Test 1000 (peak): F1 as function of -S, C% and CSF1 used for hyper-parameter selection

Chinese, Train: Brown, Test: sighan2005/as\_test\_gold 10000



[https://github.com/agents/pygents/blob/main/notebooks/nlp/chinese/zh\\_tokenizer\\_auto.ipynb](https://github.com/agents/pygents/blob/main/notebooks/nlp/chinese/zh_tokenizer_auto.ipynb)

# Chinese corpora, different “ground truth” tokenizers



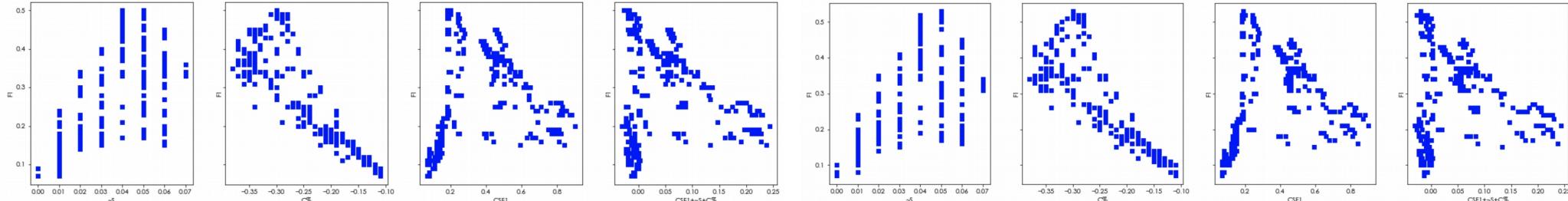
Ground truth: sighan2005

Chinese, Train: Brown, Test: sighan2005/as\_test\_gold 100

Ground truth: Jieba

Chinese Test 100 (peak): F1 as function of  $\sim S$ , C% and CSF1 used for hyper-parameter selection

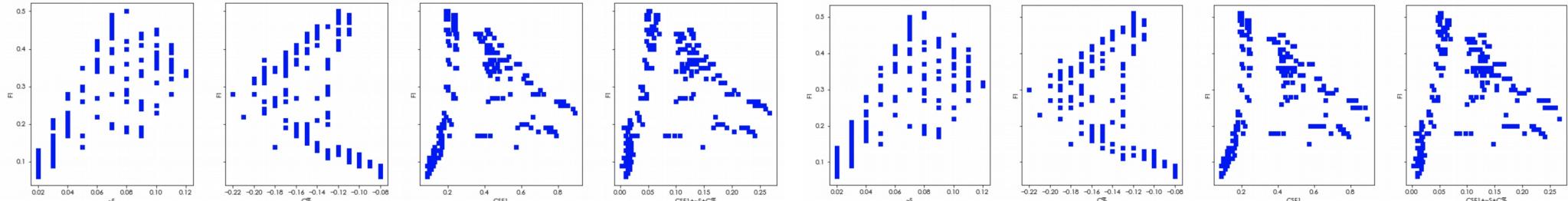
Chinese Test 100 (peak) - sighan2005/as\_test\_gold: F1 as function of  $\sim S$ , C% and CSF1 used for hyper-parameter selection



Chinese, Train: Brown, Test: sighan2005/as\_test\_gold 1000

Chinese Test 1000 (peak): F1 as function of  $\sim S$ , C% and CSF1 used for hyper-parameter selection

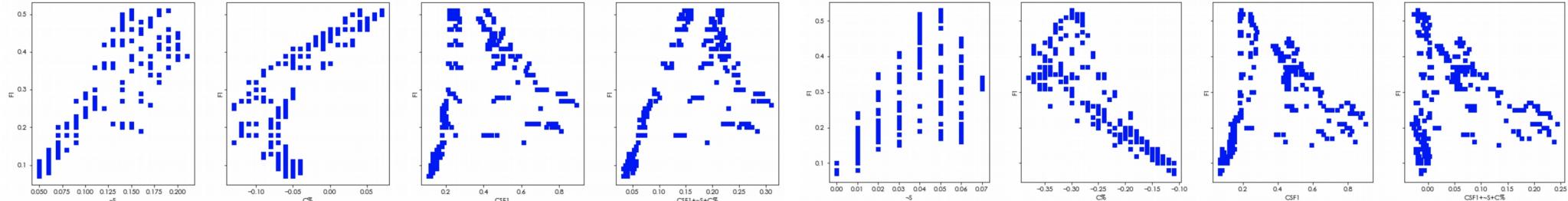
Chinese Test 1000 (peak) - sighan2005/as\_test\_gold: F1 as function of  $\sim S$ , C% and CSF1 used for hyper-parameter selection



Chinese, Train: Brown, Test: sighan2005/as\_test\_gold 10000

Chinese Test 10000 (peak): F1 as function of  $\sim S$ , C% and CSF1 used for hyper-parameter selection

Chinese Test 100 (peak) - sighan2005/as\_test\_gold: F1 as function of  $\sim S$ , C% and CSF1 used for hyper-parameter selection



[https://github.com/aignts/pygents/blob/main/notebooks/nlp/chinese/zh\\_tokenizer\\_multi-criteria-cws.ipynb](https://github.com/aignts/pygents/blob/main/notebooks/nlp/chinese/zh_tokenizer_multi-criteria-cws.ipynb)

[https://github.com/aignts/pygents/blob/main/notebooks/nlp/chinese/zh\\_tokenizer\\_multi-criteria-cws-jieba.ipynb](https://github.com/aignts/pygents/blob/main/notebooks/nlp/chinese/zh_tokenizer_multi-criteria-cws-jieba.ipynb)

# Thank You and Welcome!

Anton Kolonin

[akolonin@aigents.com](mailto:akolonin@aigents.com)

Facebook: akolonin

Telegram: akolonin



**N\***Novosibirsk  
State  
University  
**\*THE REAL SCIENCE**

<https://www.nsu.ru/>



<https://agirussia.org>