Using Probabilistic Context-Free Grammar to Create Password-Guessing Models

Isabelle Hjelden
University of Minnesota Morris

Introduction

- Text-based password
  - Most common form of authentication
  - Passwords are reused or use common patterns and word
- Database leaks or hacks
- Guessing models
  - Data-driven
  - Exploits regualities seen in samples
- Probabilistic Context-Free Grammar Guessing Models
  - Are they efficient?

babygirl
purple
1995
321
2003
1993
1989
100
alex
jordan
1234567
111
red
password
2009
2008
michael
2004
1992
qwerty
ashley
1997
123456789
4ever
2005
sunshine
123
lovely
654321
iluv
jan
boy
abc123
big
daniel
2007
mar
monkey
soccer
4life
anthony
jay
123456
man
lil
the
12345678
1994
2006
luv
777
michelle
ash
1987
and
ilove
princess
1991
rockyou
jessica
password1
friends
angel
sam
nicole
456
101
420
007
1986
1988
chocolate
2000
143
12345
car
iloveyou
000000
tigger
111111
sexy
1990
pink
1996
baby
love
1234
john
666
may
girl
teamo
1985
iloveu
forever
dan

Source: Xu et al, 2021

Outline

- Background
    - Password data leaks
    - Probabilistic context-free grammar
    - Other password cracking models
- Semantic PCFG
    - Definition
    - Password modeling example
    - Testing and results
- Chunk-Level PCFG
    - Definition
    - Password modeling example
    - Testing and results
- Conclusion

Outline

- Background
  - Password data leaks
  - Probabilistic context-free grammar
  - Other password cracking models
- Semantic PCFG
  - Definition
  - Password modeling example
  - Testing and results
- Chunk-Level PCFG
  - Definition
  - Password modeling example
  - Testing and results
- Conclusion

Background - Password Data Leaks (English)

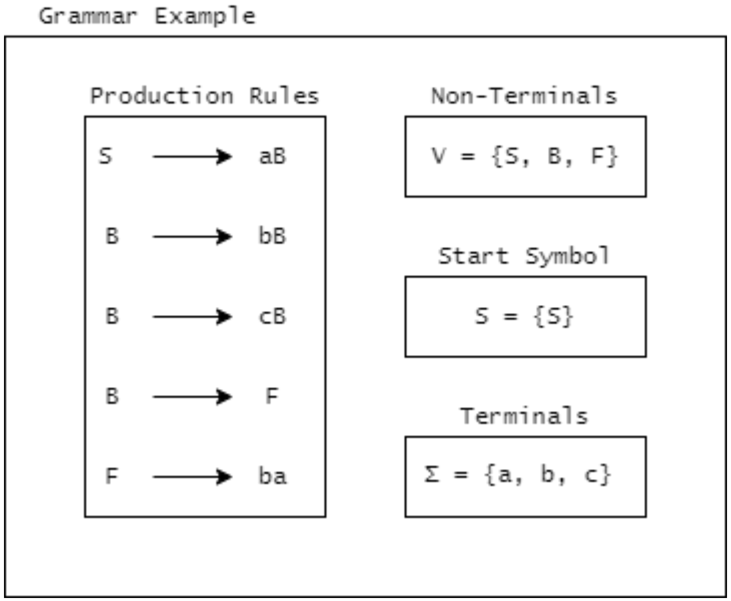
- RockYou
    - Data breach in 2009
    - 32 million passwords
- LinkedIn
    - Originally hacked in 2012, more information was released in 2016
    - 162 million passwords
- 000webhost
    - Hacked in 2015
    - 13 million passwords
- Cit0day
    - Data breach in 2020
    - 200 million passwords

Background - Password Data Leaks (Chinese)

- CSDN
    - Hacked in 2011
    - 6 million passwords
- 178
    - Hacked in 2011
    - 9 million passwords

Background - Context-Free Grammar

- Generates strings from given language
  - Sentences from English language



Grammar Example

Production Rules

$S \rightarrow aB$

$B \rightarrow bB$

$B \rightarrow cB$

$B \rightarrow F$

$F \rightarrow ba$

Non-Terminals

$V = \{S, B, F\}$

Start Symbol

$S = \{S\}$

Terminals

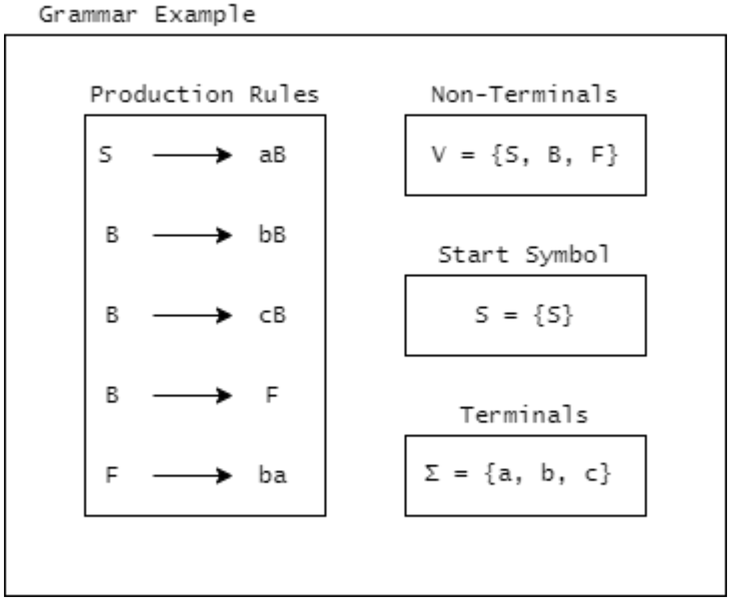
$\Sigma = \{a, b, c\}$

Image source

Background - Context-Free Grammar

- Generates strings from given language
  - Sentences from English language
- Example:

String	PR
S → aB	
→ abB	(B → bB)
→ abcB	(B → cB)
→ abcbB	(B → bB)
→ abcbF	(B → F)
→ abcbba	(F → ba)



Grammar Example

Production Rules

S → aB

B → bB

B → cB

B → F

F → ba

Non-Terminals

V = {S, B, F}

Start Symbol

S = {S}

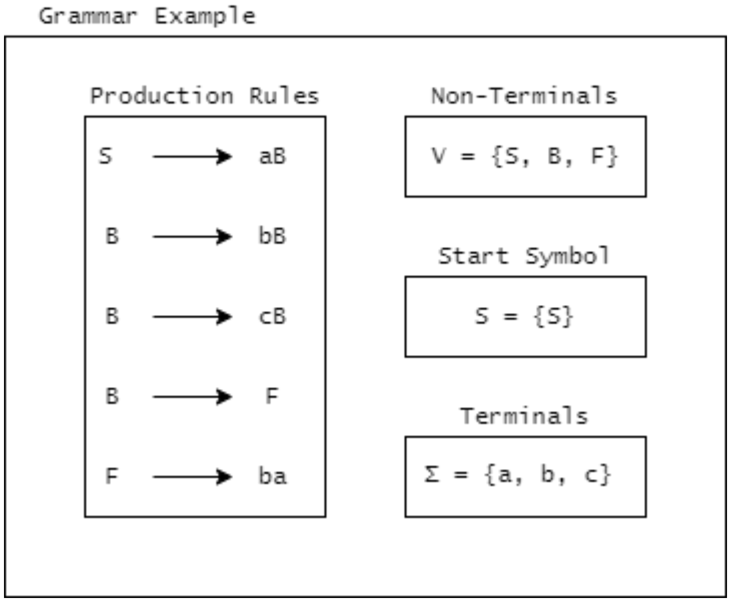
Terminals

Σ = {a, b, c}

Image source

Background - Probabilistic Context-Free Grammar

- Extension of Context-Free Grammar
- Adds probability factor to production rules
- Production rules and probability are determined by a training data set



Grammar Example

Production Rules

S → aB

B → bB

B → cB

B → F

F → ba

Non-Terminals

V = {S, B, F}

Start Symbol

S = {S}

Terminals

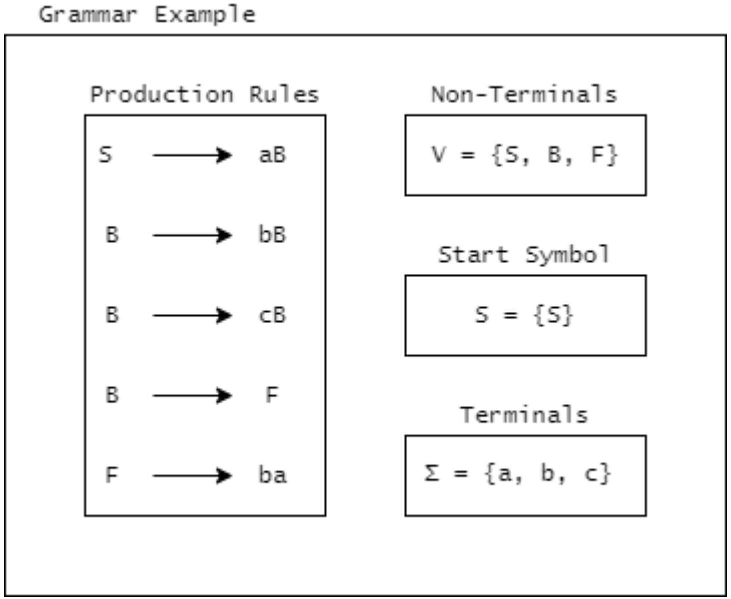
Σ = {a, b, c}

Image source

Background - Probabilistic Context-Free Grammar

- Example:

String	PR	Probability
S → aB		1.00
→ abB	(B → bB)	0.50
→ abcB	(B → cB)	0.25
→ abcbB	(B → bB)	0.50
→ abcbF	(B → F)	0.25
→ abcbba	(F → ba)	1.00



Grammar Example

Production Rules

S → aB

B → bB

B → cB

B → F

F → ba

Non-Terminals

V = {S, B, F}

Start Symbol

S = {S}

Terminals

Σ = {a, b, c}

Image source

Background - Password Cracking Models

- Controls models for testing the efficiency
  - **First PCFG model** (Weir et al.)
    - Breaks down passwords in to character classes
    - Does not use word segmentation
  - **Enhanced PCFG** (Komanduri)
    - Word segmentation
    - Learn fulls passwords as terminals
  - **Neural Network** (Melicher et al)
    - Long short-term memory

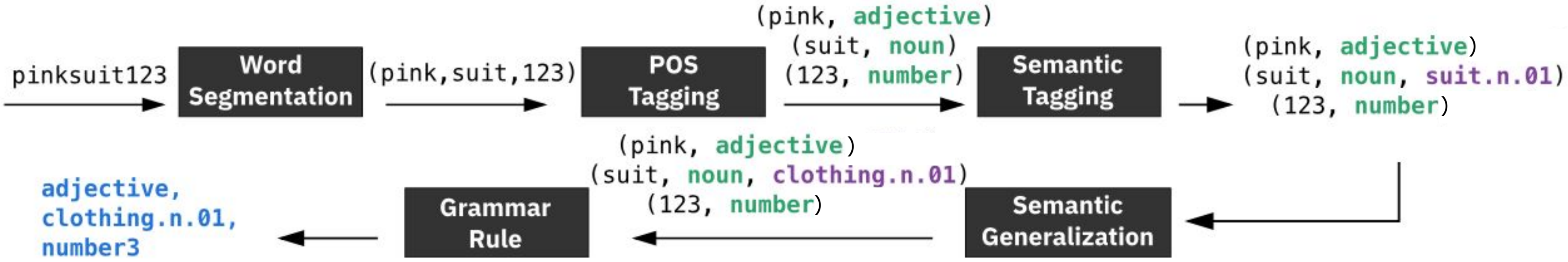
Outline

- Background
  - Password data leaks
  - Probabilistic context-free grammar
  - Other password cracking models
- Semantic PCFG
  - Definition
  - Password modeling example
  - Testing and Results
- Chunk-Level PCFG
  - Definition
  - Password modeling example
  - Testing and Results
- Conclusion

Semantic Definition

- Parts-of-speech and semantics
- Training the grammar
    - Text processing pipeline
    - Semantic generalization
    - Does not classify misspellings or substitutions
- Probability
    - Maximum length estimation (MLE)
        - The more frequent a production rule is seen, the higher the probability
    - Terminal smoothing to deter overfitting
        - Laplace formula

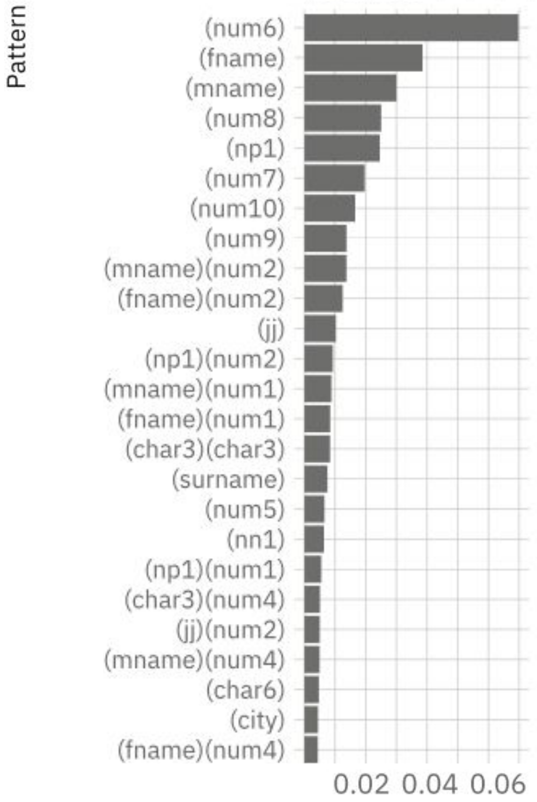
Semantic - Password Modeling Example



Source: Veras et al, 2021

Semantic - Password Modeling

- Top grammar rules from RockYou
- Parts of speech tags (CLAWS7)
  - **np** - singular proper noun
  - **jj** - adjective
  - **mname**/**fname** - male/female name
  - **char** - unidentified words or symbols
  - **num** - number
  - **-#** - amount of objects
    - Ex: (num6) = 123456, 132436

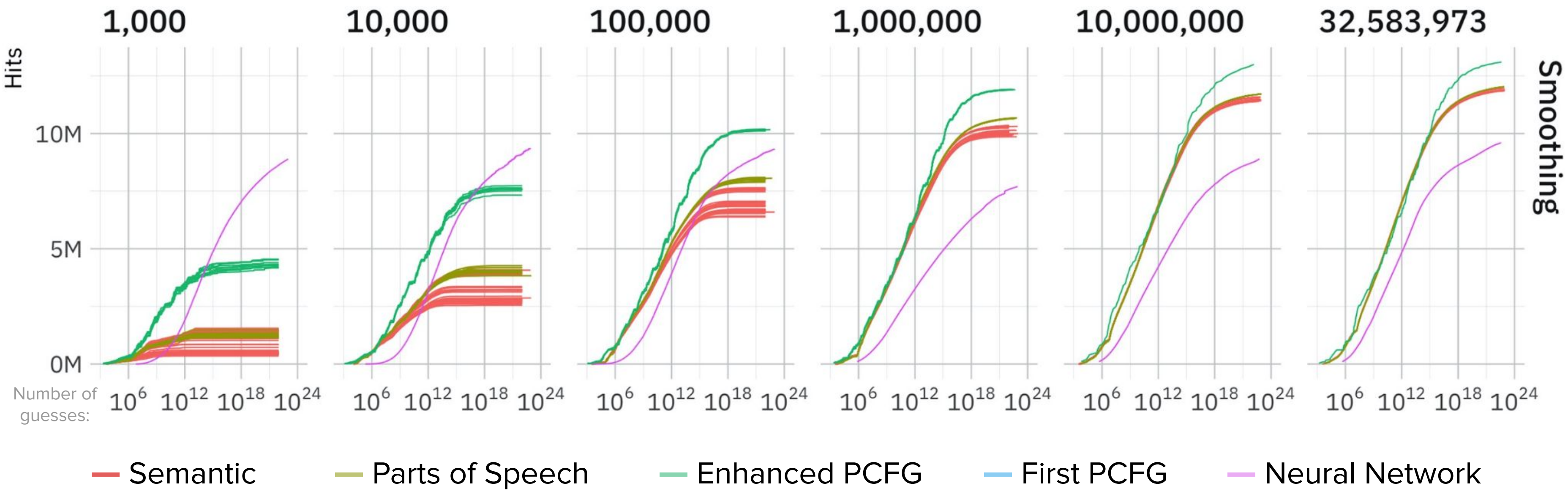


Source: Veras et al, 2021

Semantic - Testing

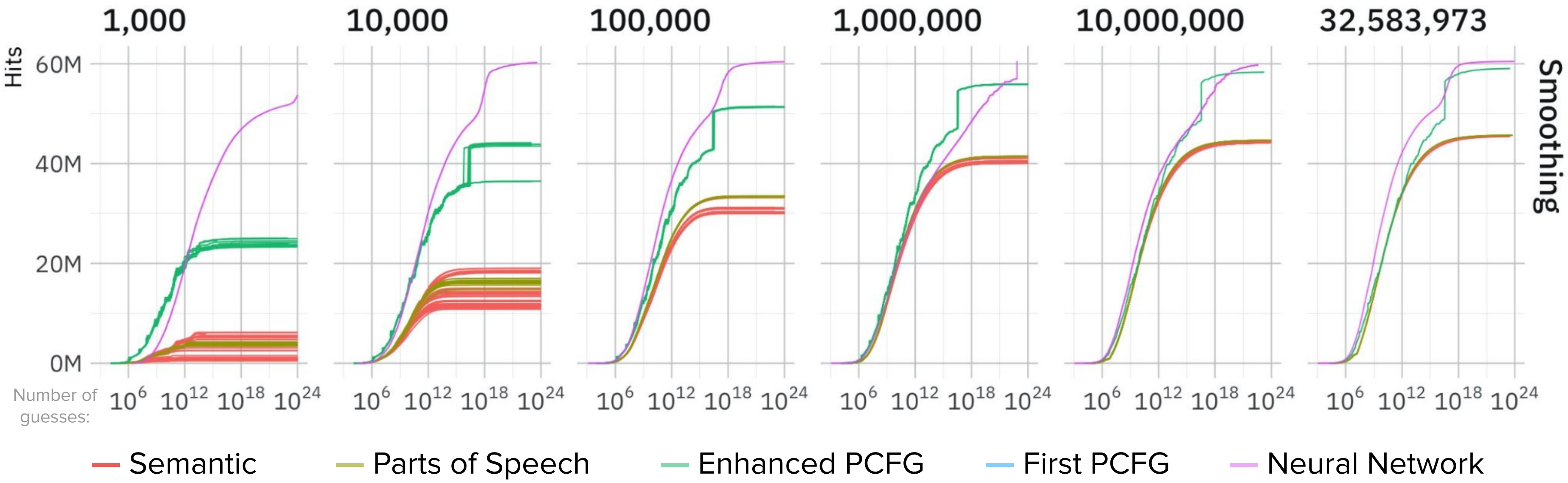
- Models trained on multiple size samples
  - 1,000 | 10,000 | 100,000 | 1,000,000 | 10,000,000 | 32,583,973
- Tested the model with different levels semantic accuracy
- Tested the model without semantic tagging (parts of speech)
- Tested the model with and without terminal smoothing

Semantic - Results from 000webhost



Source: Veras et al, 2021

Semantic - Results from LinkedIn



Source: Veras et al, 2021

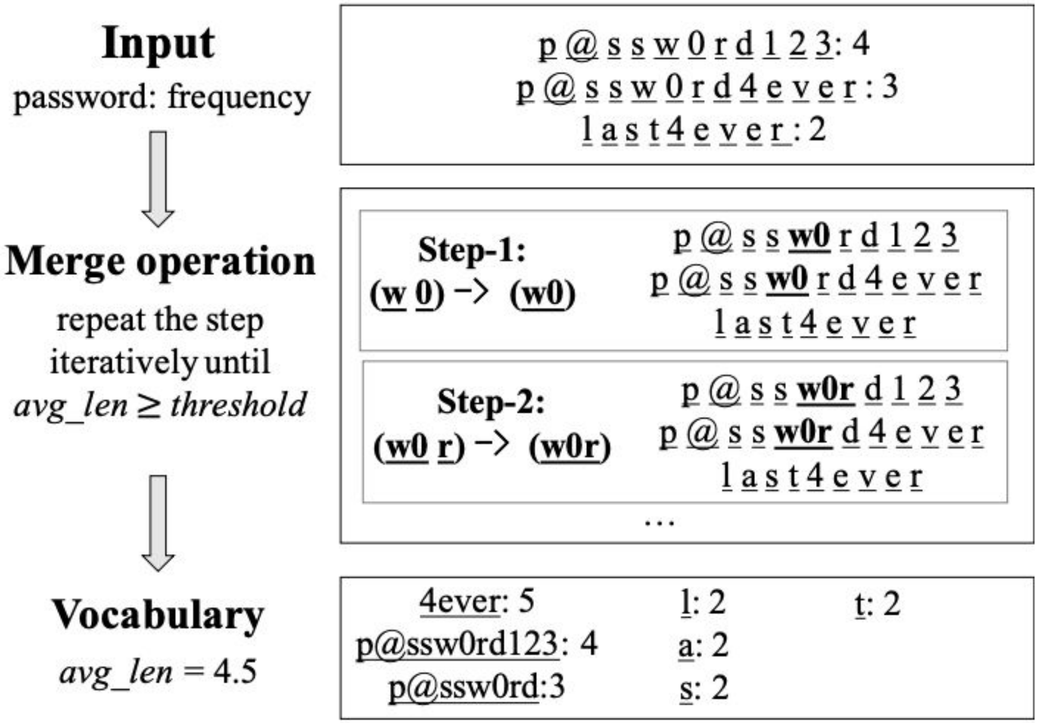
Outline

- Background
  - Password data leaks
  - Probabilistic context-free grammar
  - Other password cracking models
- Semantic PCFG
  - Definition
  - Password modeling example
  - Testing and Results
- Chunk-Level PCFG
  - Definition
  - Password modeling example
  - Testing and Results
- Conclusion

Chunk-Level - Definition

- Password specific segmentation
  - Extend the Byte-Pair-Encoding algorithm
  - Merges character pairs, then creates vocabulary
- Probability
  - Maximum length estimation

Chunk-Level - Password Modeling Example



Input

password: frequency

p @ s s w 0 r d 1 2 3: 4
p @ s s w 0 r d 4 e v e r : 3
l a s t 4 e v e r : 2

Merge operation

repeat the step iteratively until *avg_len* ≥ *threshold*

Step-1:
(w 0) -> (w0)

p @ s s **w0** r d 1 2 3
p @ s s **w0** r d 4 e v e r
l a s t 4 e v e r

Step-2:
(w0 r) -> (w0r)

p @ s s **w0r** d 1 2 3
p @ s s **w0r** d 4 e v e r
l a s t 4 e v e r

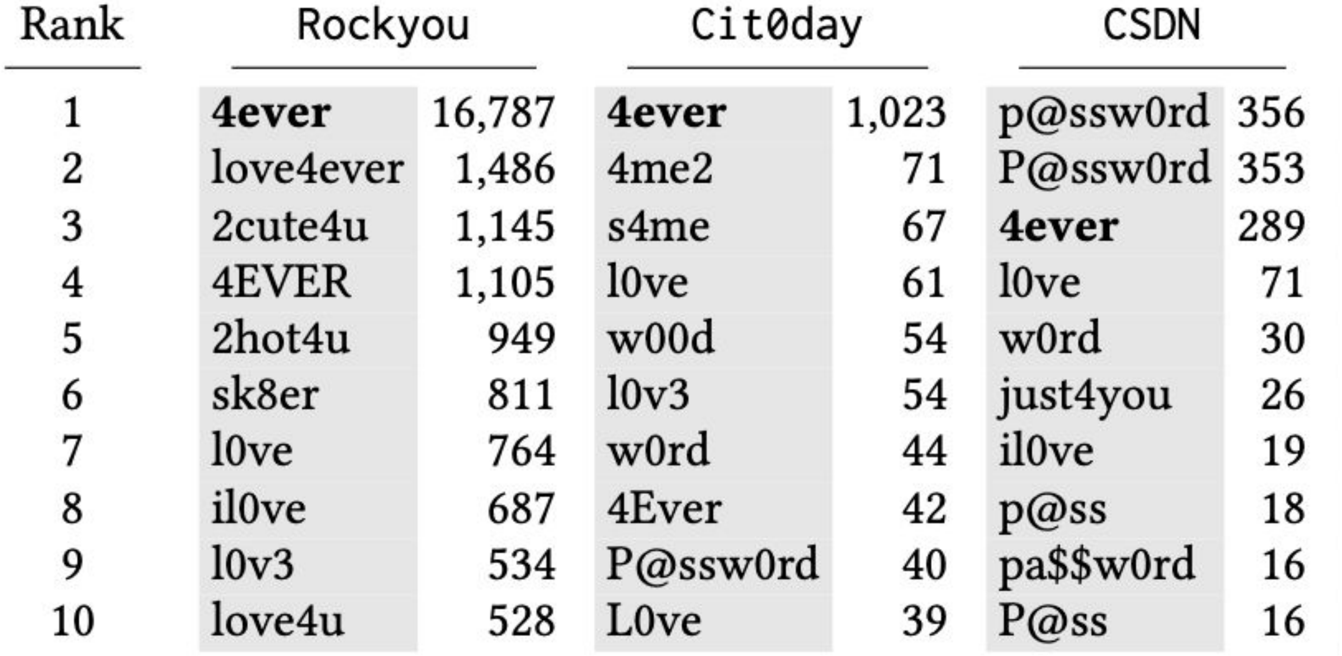
…

Vocabulary

avg_len = 4.5

4ever: 5	l: 2	t: 2
p@ssw0rd123: 4	a: 2	
p@ssw0rd:3	s: 2	

Chunk-Level - Password Modeling



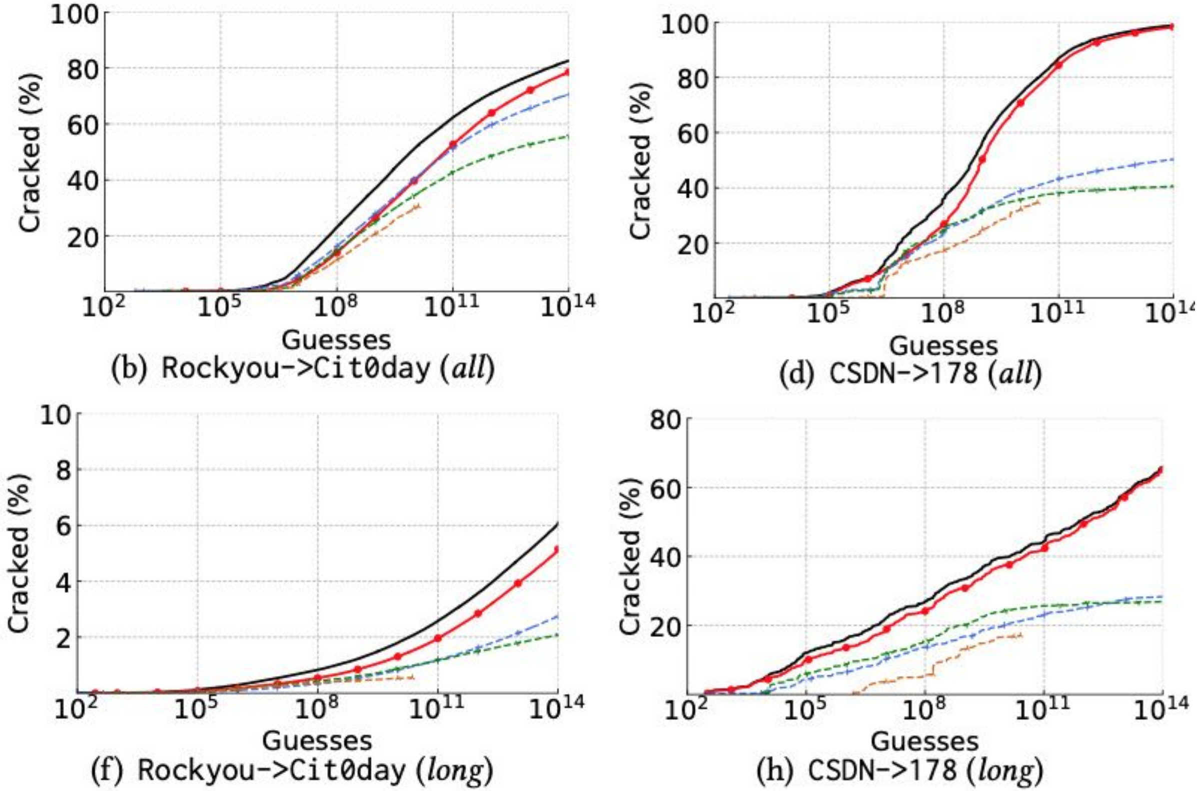
Rank	Rockyou		Cit0day		CSDN	
1	**4ever**	16,787	**4ever**	1,023	p@ssw0rd	356
2	love4ever	1,486	4me2	71	P@ssw0rd	353
3	2cute4u	1,145	s4me	67	**4ever**	289
4	4EVER	1,105	l0ve	61	l0ve	71
5	2hot4u	949	w00d	54	w0rd	30
6	sk8er	811	l0v3	54	just4you	26
7	l0ve	764	w0rd	44	il0ve	19
8	il0ve	687	4Ever	42	p@ss	18
9	l0v3	534	P@ssw0rd	40	pa$$w0rd	16
10	love4u	528	L0ve	39	P@ss	16

Top chunks with misspellings or substitutions

Chunk-Level - Testing

- Trained on English and Chinese passwords
  - English passwords from RockYou leak
  - Chinese passwords from CSDN leak
- Models were ran on two samples
  - First test: all passwords leaked from Cit0day and 178
  - Second test: passwords equal or longer to 16 characters from Cit0day and 178

Chunk-Level - Results from Cit0day and 178



(b) Rockyou->Cit0day (*all*)

(d) CSDN->178 (*all*)

(f) Rockyou->Cit0day (*long*)

(h) CSDN->178 (*long*)

— Chunk-Level PCFG

- - Semantic PCFG

- - Enhanced PCFG

- - First PCFG

— Min_auto

-(not a control model)

Source: Xu et al, 2021

Outline

- Background
  - Password data leaks
  - Probabilistic context-free grammar
  - Other password cracking models
- Semantic PCFG
  - Definition
  - Password modeling example
  - Testing and Results
- Chunk-Level PCFG
  - Definition
  - Password modeling example
  - Testing and Results
- Conclusion

Conclusion

- Newer PCFG models are becoming better at guessing passwords
- PCFG models are intended to identify weak passwords
  - Helps companies and users create stronger passwords
- The data leaks used to train and test these models are outdated

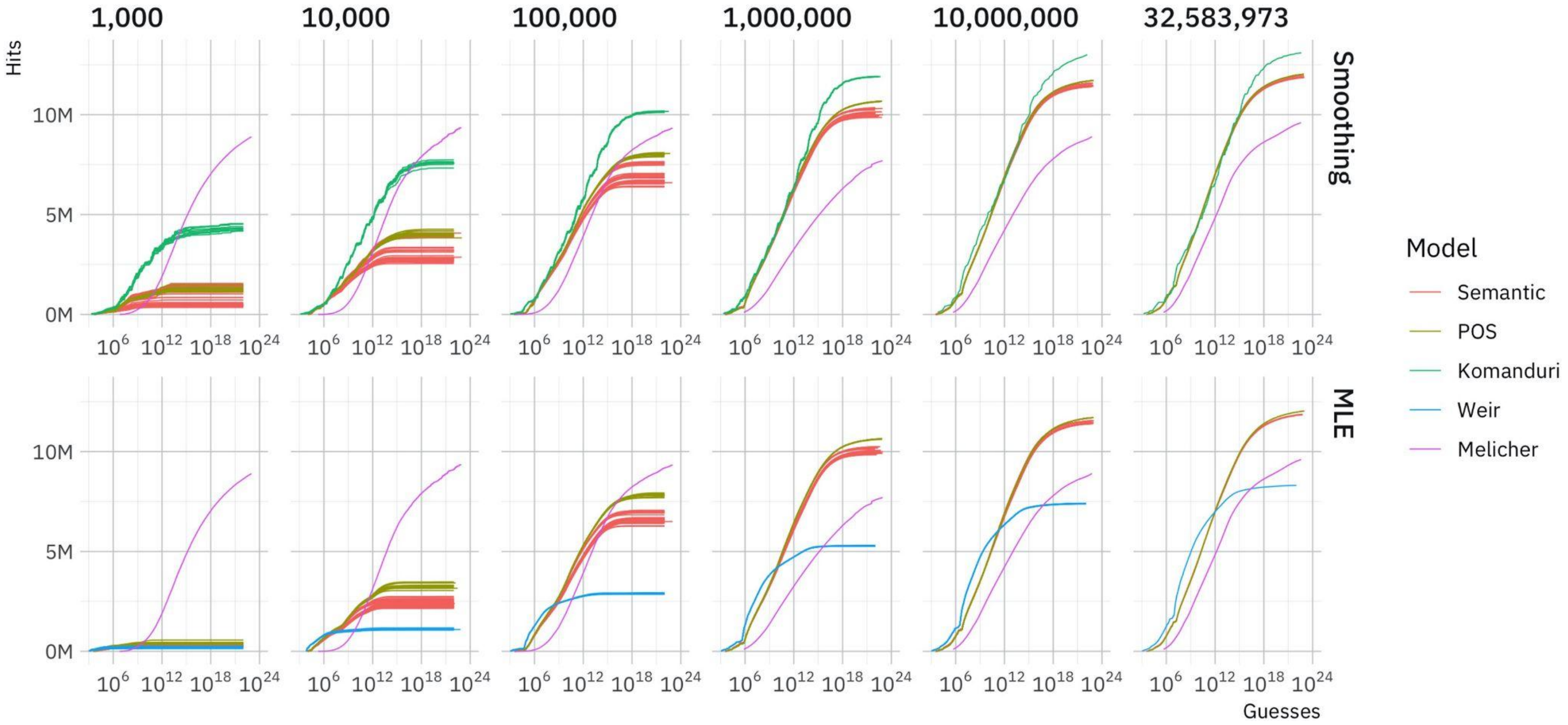
References

Rafael Veras, Christopher Collins, and Julie Thorpe. 2021. A Large-Scale Analysis of the Semantic Password Model and Linguistic Patterns in Passwords. ACM Trans. Priv. Secur. 24, 3, Article 20 (apr 2021), 21 pages. https://doi.org/10.1145/3448608

Ming Xu, Chuanwang Wang, Jitao Yu, Junjie Zhang, Kai Zhang, and Weili Han. 2021. Chunk-Level Password Guessing: Towards Modeling Refined Password Composition Representations. In Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security (Virtual Event, Republic of Korea) (CCS ’21). Association for Computing Machinery, New York, NY, USA, 5–20. https://doi.org/10.1145/3460120.3484743

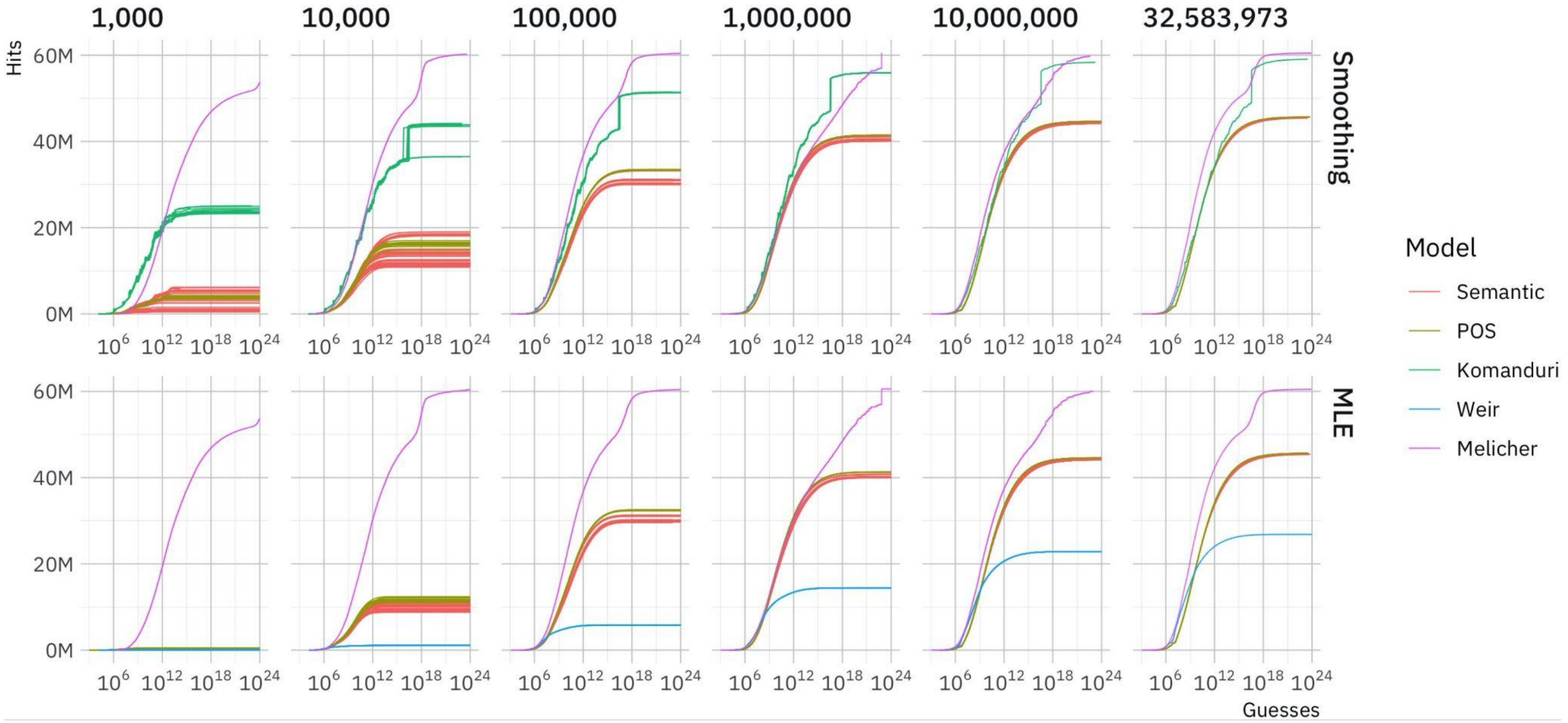
Questions?

Complete Results - Semantic PCFG 000webhost



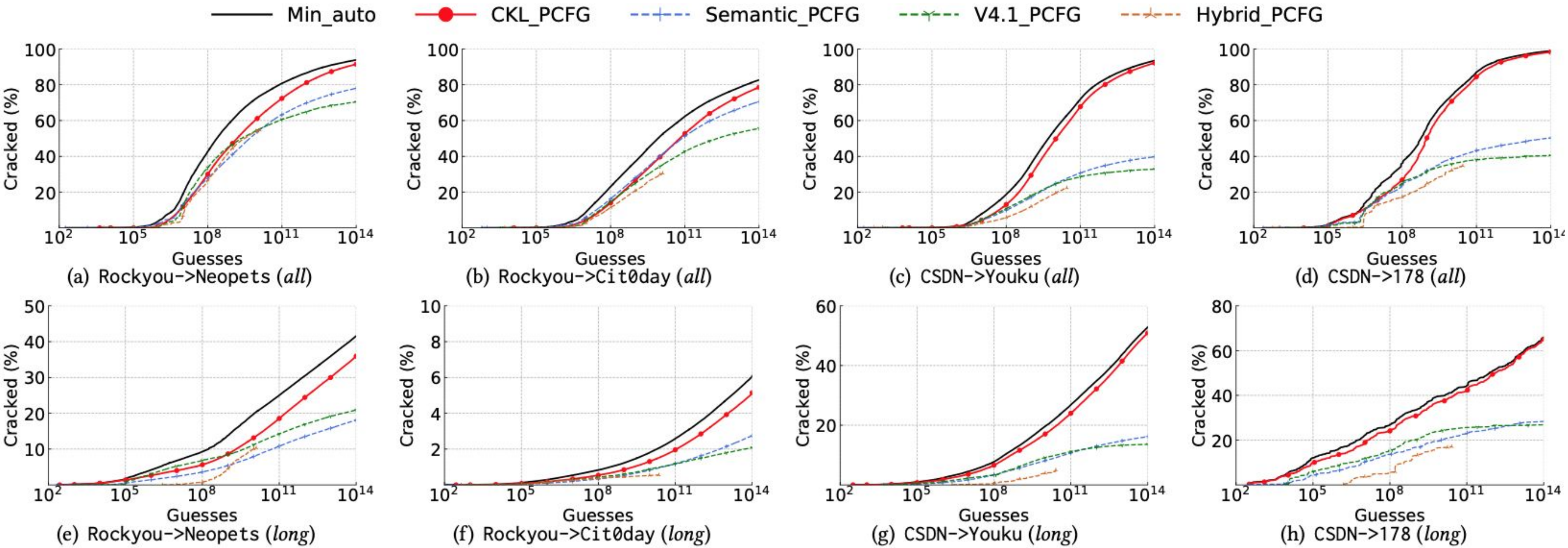
Source: Veras et al, 2021

Complete Results - Semantic PCFG LinkedIn



Source: Veras et al, 2021

Complete Results - Chunk level PCFG



(a) Rockyou->Neopets (*all*)
(b) Rockyou->Cit0day (*all*)
(c) CSDN->Youku (*all*)
(d) CSDN->178 (*all*)
(e) Rockyou->Neopets (*long*)
(f) Rockyou->Cit0day (*long*)
(g) CSDN->Youku (*long*)
(h) CSDN->178 (*long*)

Source: Xu et al, 2021