Authorship Analysis of Online Predators using Character Level Convolutional Neural Networks

Kanishka Misra, Hemanth Devarapalli, Tatiana Ringenberg, Julia Taylor Rayz

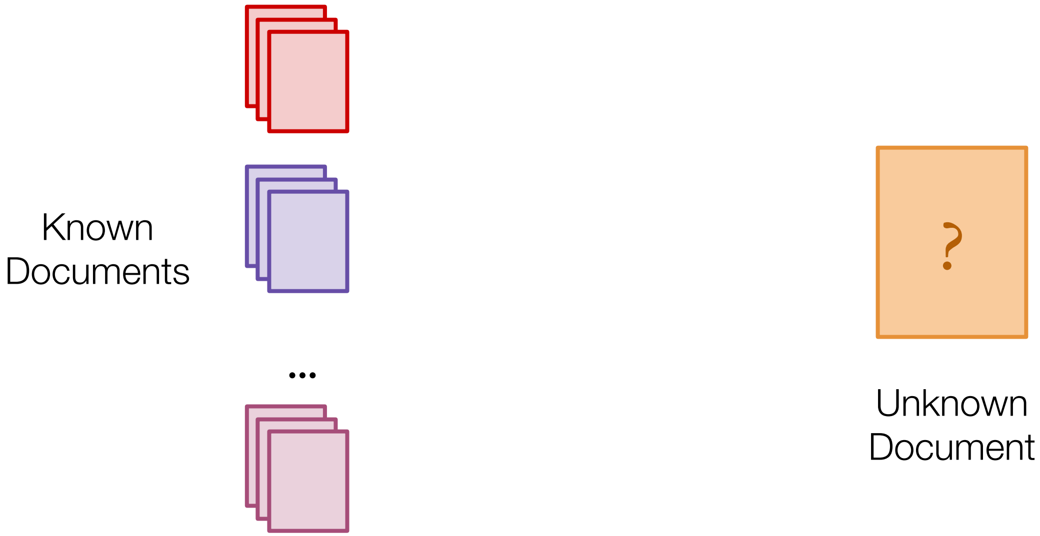
Applied Knowledge Representation and Natural Language Understanding Lab
Purdue University

Authorship Attribution

Assigning an author to a piece of text whose author is unknown.

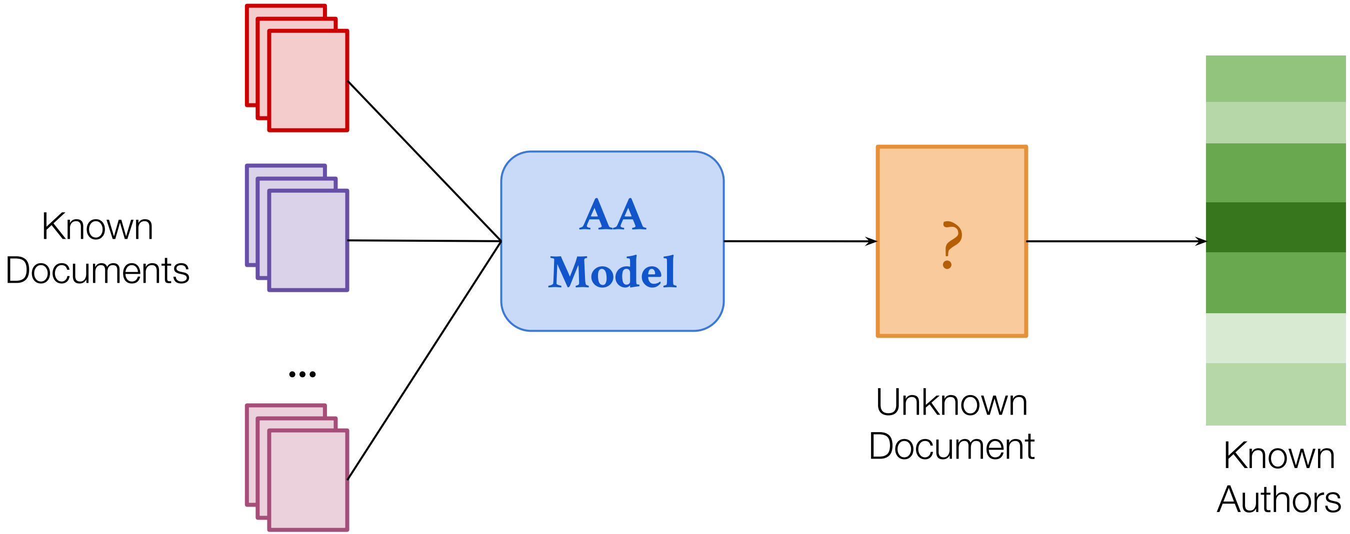
Authorship Attribution

Assigning an author to a piece of text whose author is unknown.



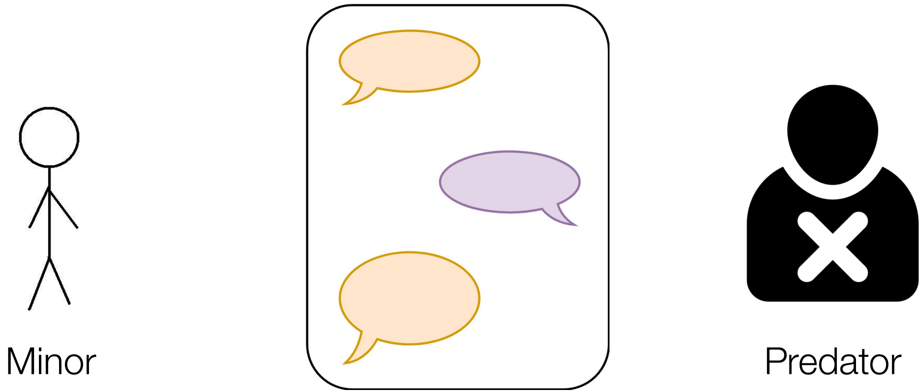
Authorship Attribution

Assigning an author to a piece of text whose author is unknown.



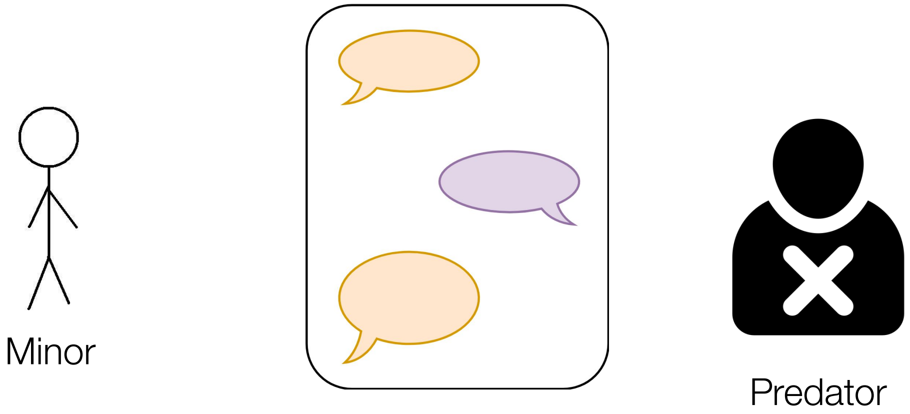
Predatory **Conversations**

The National Center for Missing and Exploited Children (NCMEC) received 10.2 million reports of suspected child exploitation in 2017.



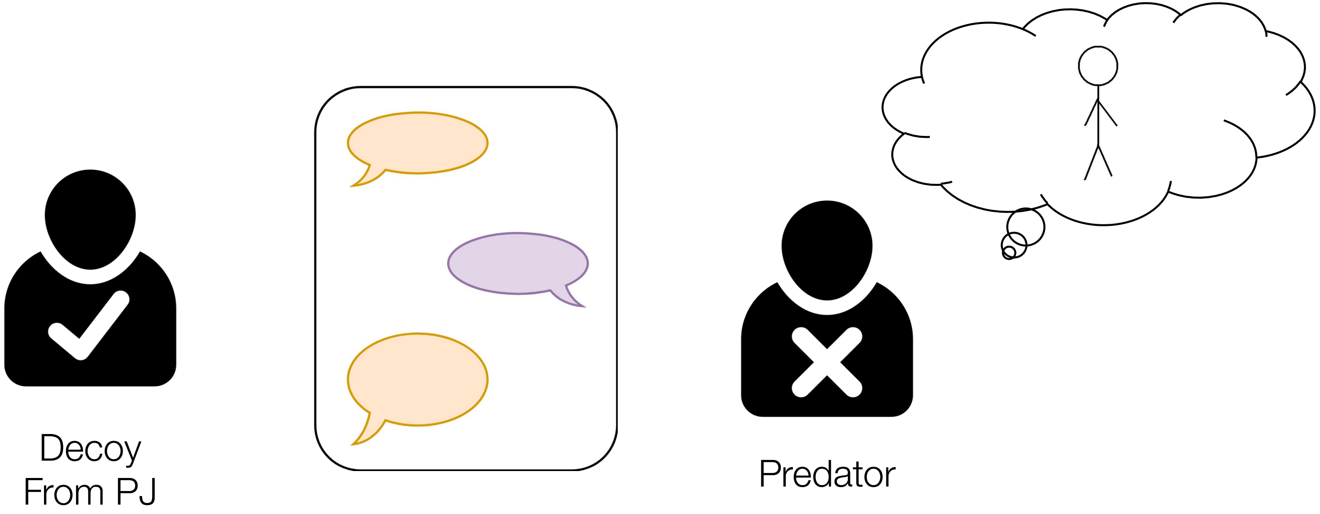
Icon made by Freepik from www.flaticon.com

The Perverted Justice Corpus



Icon made by Freepik from www.flaticon.com

The Perverted Justice Corpus



Icons made by Freepik from www.flaticon.com

The Perverted Justice Corpus

1. Vigilante organization which helps law enforcement perform sting operations
2. Website stores conversations between offenders and decoys
3. Decoys pretend to be a minor
4. 2004 to present
5. 623 chats

Research Objectives

Given chat conversations between predators and decoys, and between regular people:

1. **Can we successfully identify the author of unknown chat lines?** (Comparable to State of the Art).
2. **Can we separate predators from non-predators using the encoded message representation that is trained to *only learn* the author's style?**

Research Contributions

1. Place online predators in an Authorship Attribution/Analysis Framework.
2. Propose two new models that operate at the state of the art level for short text AA (for our dataset).
    a. **AA-CNN:** A Character Level CNN that is trained to only do AA.
    b. **AA-CNN-PC:** A Character Level CNN that is jointly trained to do AA as well as to distinguish between predators and non-predators.
3. Propose a test that analyzes the properties of the Chat Message Representations
    a. Does a model that is only trained to *learn* author style also differentiate between the type of author?

Capturing the Author's Style

1. Traditionally:
    a. **Lexical:** tf/tf-idf of word/character n-grams used in documents, k-signatures, only functional words
    b. **Syntactical:** POS Tags, Dependency Relations.
    c. **Misc:** Sentence length, whitespaces, etc.
2. Character n-grams have been found to be very robust!
3. Idea is to get an 'author vector' of some sort to feed to a classifier.

(Koppel and Schler, 2003; Argamon et al. 2007; Stamatatos, 2009; Koppel et al. 2011; Schwartz et al. 2013)

Capturing the Author's Style

1. Traditionally:
   a. **Lexical:** tf/tf-idf of word/character n-grams used in documents, k-signatures, only functional words
   b. **Syntactical:** POS Tags, Dependency Relations.
   c. **Misc:** Sentence length, whitespaces, etc.
2. Character n-grams have been found to be very robust!
3. Idea is to get an 'author vector' of some sort to feed to a classifier.

Work well for Long Documents!

(Koppel and Schler, 2003; Argamon et al. 2007; Stamatatos, 2009; Koppel et al. 2011; Schwartz et al. 2013)

Capturing the Author's Style - *Short Texts*

1. Two paths:
   a. Take each text separately
   b. Bundle chunks of short texts together into a document
2. Both result in sparse vectors if we use count based features
3. Dense representations - sentence encoders (CNN, LSTMs, etc.)
4. Literature: Character sequence + CNN.

(Ruder et al., 2016; Sari et al., 2017; Shrestha et al., 2017)

Experimental Setup

1. **Corpus:** Perverted Justice + PAN 2012 Corpus (IRC Chat logs, regular conversations).

Experimental Setup

1. **Corpus:** Perverted Justice + PAN 2012 Corpus (IRC Chat logs, regular conversations).
2. Coalesce same-author messages that are one after the other.

Experimental Setup

1. **Corpus:** Perverted Justice + PAN 2012 Corpus (IRC Chat logs, regular conversations).
2. Coalesce same-author messages that are one after the other.
3. Filtered for authors that have at least 600 unique lines. **345 Authors.**

Experimental Setup

1. **Corpus:** Perverted Justice + PAN 2012 Corpus (IRC Chat logs, regular conversations).
2. Coalesce same-author messages that are one after the other.
3. Filtered for authors that have at least 600 unique lines. **345 Authors.**
4. Sample sets of:
   a. 10 authors (5 predators, 5 regular users)
   b. 50 authors (25 predators, 25 regular users)
   c. **Remain Consistent with Prior Work!**

Experimental Setup

1. **Corpus:** Perverted Justice + PAN 2012 Corpus (IRC Chat logs, regular conversations).
2. Coalesce same-author messages that are one after the other.
3. Filtered for authors that have at least 600 unique lines. **345 Authors.**
4. Sample sets of:
   a. 10 authors (5 predators, 5 regular users)
   b. 50 authors (25 predators, 25 regular users)
   c. **Remain Consistent with Prior Work!**
5. For each set, sample 400, 100, 100 chat lines ***per author*** as train, validation, and test sets.

Experimental Setup

1. **Corpus:** Perverted Justice + PAN 2012 Corpus (IRC Chat logs, regular conversations).
2. Coalesce same-author messages that are one after the other.
3. Filtered for authors that have at least 600 unique lines. **345 Authors.**
4. Sample sets of:
   a. 10 authors (5 predators, 5 regular users)
   b. 50 authors (25 predators, 25 regular users)
   c. **Remain Consistent with Prior Work!**
5. For each set, sample 400, 100, 100 chat lines ***per author*** as train, validation, and test sets.
6. Train Models!

Experimental Setup

1. **Corpus:** Perverted Justice + PAN 2012 Corpus (IRC Chat logs, regular conversations).
2. Coalesce same-author messages that are one after the other.
3. Filtered for authors that have at least 600 unique lines. **345 Authors.**
4. Sample sets of:
   a. 10 authors (5 predators, 5 regular users) **4000/1000/1000**
   b. 50 authors (25 predators, 25 regular users) **20000/5000/5000**
   c. **Remain Consistent with Prior Work!**
5. For each set, sample 400, 100, 100 chat lines ***per author*** as train, validation, and test sets.
6. Train Models!

Baselines/Benchmarks

- Ruder et al. 2016
  - Embedding Size: 300
  - Unigram Character Level CNN.
  - Window sizes: 6, 7, 8
  - Feature Maps: 100
  - Stochastic Gradient Descent with Adadelta
  - 15 Epochs
  - Best results on AA for Tweets, Emails and Reddit comments (10 and 50 authors, 2016)
- Shrestha et al. 2017
  - Embedding Size: 300
  - 2 Models
    - Unigram Level CNN
    - Bigram Level CNN
  - Window sizes: 3,4,5
  - Feature Maps: 500
  - Adam Optimizer
  - 100 Epochs
  - Best Results on Tweets (10 and 50 authors, 2017).

Convolutional Neural Network Refresher

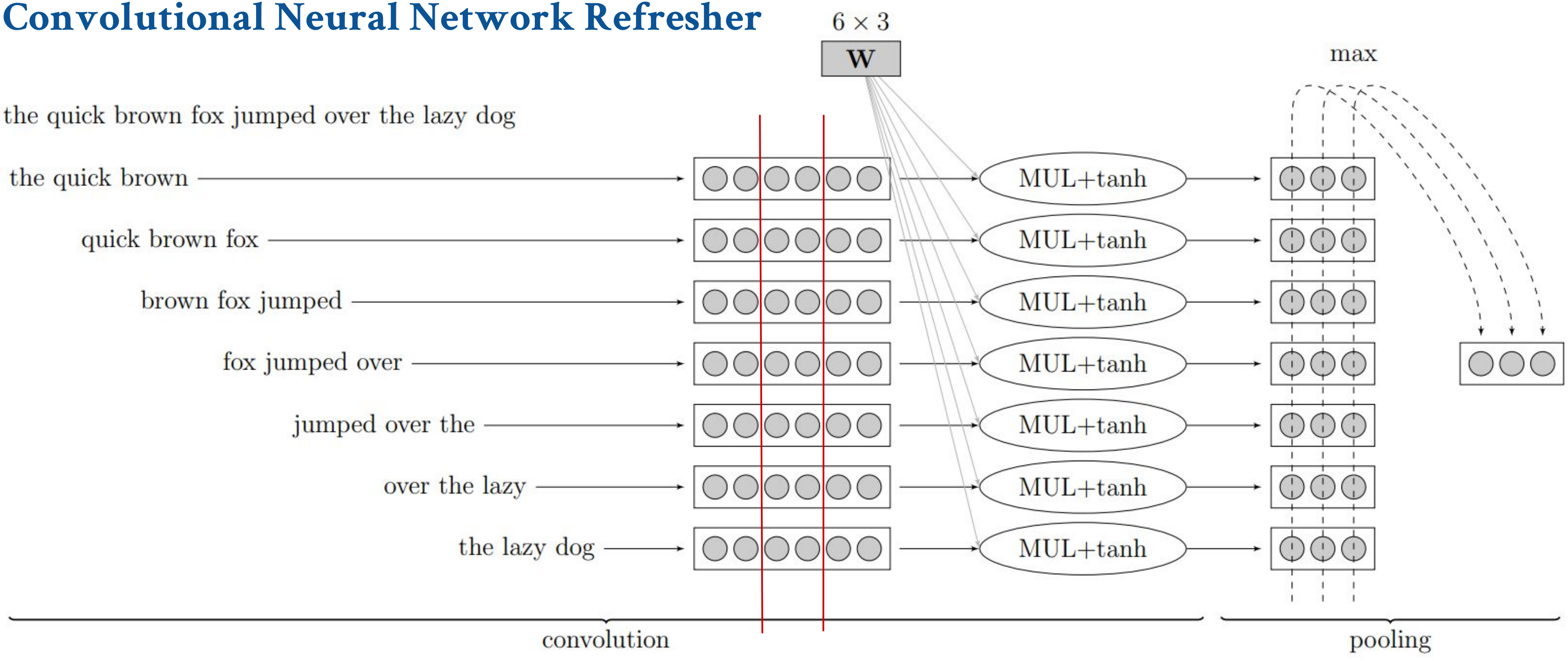
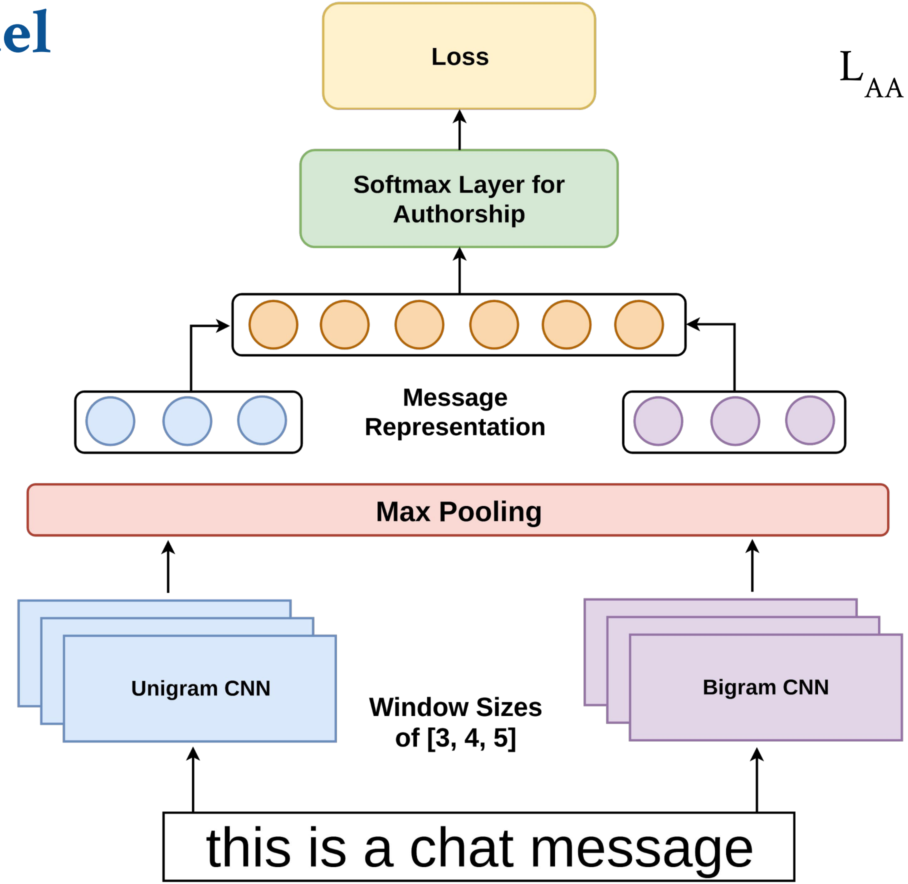


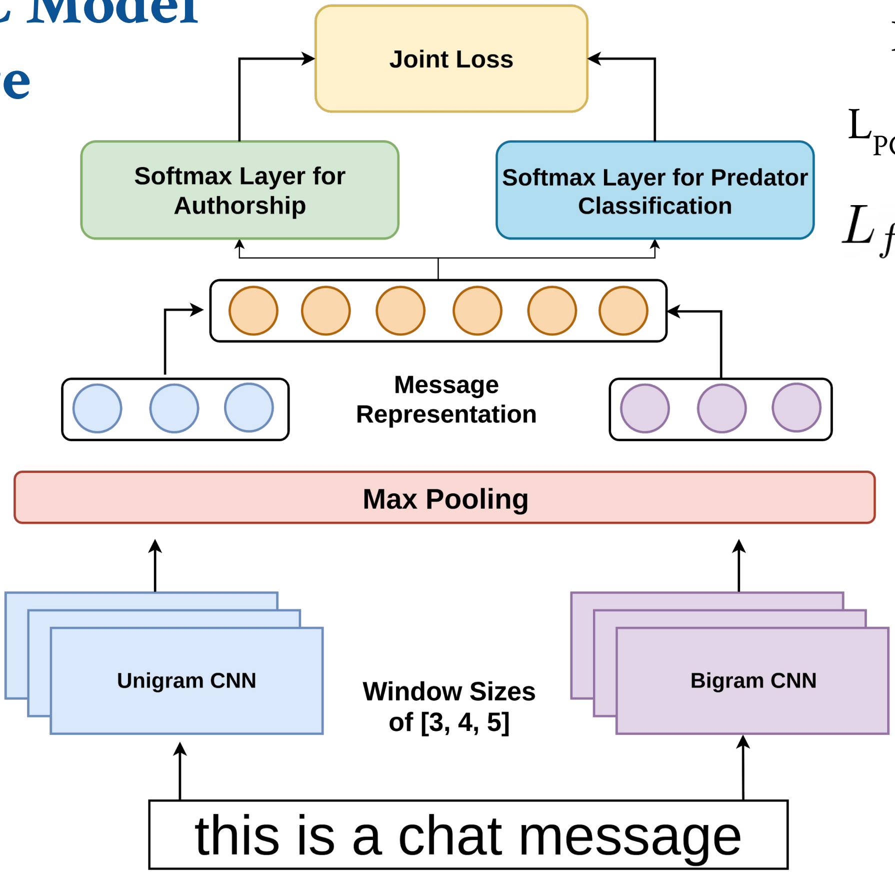
Figure Source: *A Primer on Neural Network Models for Natural Language Processing,* Yoav Goldberg

AA-CNN Model Architecture



L_{AA} = Cross Entropy

AA-CNN-PC Model Architecture



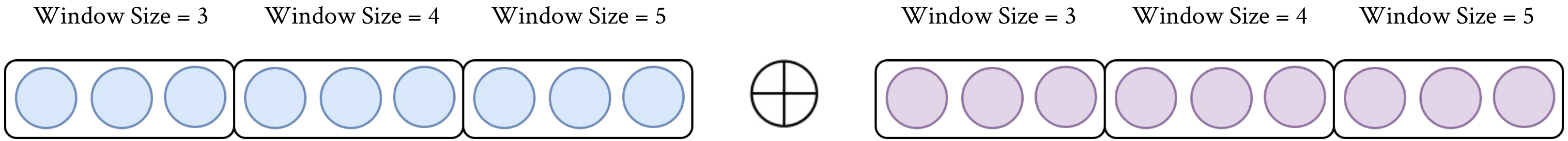
L_{AA} = Cross Entropy

L_{PC} = Binary Cross Entropy

$$L_{final} = L_{AA} + L_{PC}$$

Jointly Optimized!

Message Representation



Unigram CNN max pooled output

Bigram CNN max pooled output

Training Details

- Embedding Size = 100
- Windows = [3, 4, 5]
- Filter Maps = 100
- Final Message Representation size = 3 * 100 * 2 = 600
- Softmax layer hidden dimension = 200
- 50 Epochs with Mini-Batch Size = 32
- Adam Optimizer with learning rate of 0.001 (best out of 0.1, 0.01, 0.005)

Results

Evaluation Metric: F1 Score (Micro-Averaged)

Model	Architecture	10 Authors	50 Authors
Ruder et al. 2016	**Emb Size:** 300 **Feature Maps:** 100	0.5250	0.3524
Shrestha et al. 2017	**Emb Size:** 300 **Feature Maps:** 500	**0.5880**	0.4474
Ours (AA-CNN)	**Emb Size:** 100 x 2 **Feature Maps:** 100	0.5770	0.4382
Ours (AA-CNN-PC)		0.5490	**0.4484**

Probing Message Representations

Two Models:

1. **AA-CNN** → **Encodes Author Style only**
2. **AA-CNN-PC** → **Encodes Author Style as well as Type (Predator vs Non-Predator)**

Probing Message Representations

Two Models:

1. **AA-CNN** → **Encodes Author Style only**
2. **AA-CNN-PC** → **Encodes Author Style as well as Type (Predator vs Non-Predator)**

Q. Do the message representations learnt by AA-CNN also encode differences between predators and non-predators?

t-sne representation Encoded by AA-CNN

t-sne representation Encoded by AA-CNN-PC

Probing Message Representations - *Methodology*

Metric: **Mean Average Similarity**

$$MAS(v_i^a, v_j^b) = \frac{1}{N_i}\frac{1}{N_j}\sum_{i}^{N_i}\sum_{j}^{N_j} cos(v_i^a, v_j^b)$$

$$\Delta MAS = MAS(v_i^{predator}, v_j^{predator})1_{i \neq j} - MAS(v_i^{predator}, v_j^{non-predator})$$

Difference between the MAS of predatory messages to every other predatory message and predatory messages to every other non-predatory message.

Probing Message Representations - *Methodology*

For each model (over 10000 iterations):

1. Sample 1000 predatory messages, and 1000 non-predatory messages (with replacement).
2. Compute ΔMAS for each iteration.
3. Conduct a t-test to measure significance of ΔMAS.

Probing Message Representations - *Methodology*

For each model (over 10000 iterations):

1. Sample 1000 predatory messages, and 1000 non-predatory messages (with replacement).
2. Compute ΔMAS for each iteration.
3. Conduct a t-test to measure significance of ΔMAS.

Significant ΔMAS would indicate the model *learnt* to differentiate between predatory and non-predatory messages!

Probing Message Representations - *Results*

Model	**ΔMAS**	**Significance Test Results**
AA-CNN	**0.021**	$t = 1048.3$, $p = 2.2 \times 10^{-16}$
AA-CNN-PC	**0.025**	$t = 1285.8$, $p = 2.2 \times 10^{-16}$

Conclusion

- Presented an analysis of authorship within a predatory conversations domain.
- Developed two models:
  - **AA-CNN** → **Encodes Author Style only**
  - **AA-CNN-PC** → **Encodes Author Style as well as Type (Predator vs Non-Predator)**
- Both models were comparable to state of the art.
- Analysis of message representation found the model that encodes only stylistic properties also ***learns*** certain differentiating signals between predatory and non-predatory messages.

Conclusion

- Presented an analysis of authorship within a predatory conversations domain.
- Developed two models:
  - **AA-CNN** → **Encodes Author Style only**
  - **AA-CNN-PC** → **Encodes Author Style as well as Type (Predator vs Non-Predator)**
- Both models were comparable to state of the art.
- Analysis of message representation found the model that encodes only stylistic properties also ***learns*** certain differentiating signals between predatory and non-predatory messages.
- **However, this difference is slightly less as compared to a model that has supervised signal for both author style and author type.**

Future Work

- Tying into risk associated with the Predator in predator chats (Presenting on Wednesday, in the *fuzzy systems and their applications* session, **WeAT3**).
- Scaling up to large set of unique authors.

PURDUE
UNIVERSITY

Thank You!

Questions?

PURDUE
RESEARCH FOUNDATION

AKRaNLU

Kanishka - @iamasharkskin
Hemanth - @daemon92

kmisra@purdue.edu
hdevarap@purdue.edu
trigenb@purdue.edu
jtaylor1@purdue.edu

Coming soon..

References

1. S. Argamon et al. “Stylistic text classification using functional lexical features”. In:*Journal of the American Society for Information Science and Technology* 58.6 (2007), pp. 802–822.
2. M. Eder. “Does size matter? Authorship attribution, small samples, big problem”. In: *Digital Scholarship in the Humanities* 30.2 (Nov. 2013), pp. 167–182.ISSN: 2055-7671.
3. G. Inches and F. Crestani. “Overview of the International Sexual Predator Identification Competition at PAN-2012.” In:*CLEF (Online working notes/labs/workshop).* Vol. 30.2012
4. M. Koppel and J. Schler. “Exploiting stylistic idiosyncrasies for authorship attribution”. In: *Proceedings of IJCAI'03 Workshop on Computational Approaches to Style Analysis and Synthesis.* Vol. 69. 2003, pp. 72–80.
5. S. Ruder, P. Ghaffari, and J. G. Breslin. “Character-level and multi-channel convolutional neural networks for large-scale authorship attribution”. In:*arXiv preprint arXiv:1609.06686* (2016).
6. Y. Sari, A. Vlachos, and M. Stevenson. “Continuous n-gram representations for authorship attribution”. In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2,* Short Papers. Vol. 2. 2017, pp. 267–273
7. R. Schwartz et al. “Authorship attribution of micro-messages”. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. 2013,pp. 1880–1891.
8. P. Shrestha et al. “Convolutional neural networks for authorship attribution of short texts”. In:Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. Vol. 2.2017, pp. 669–674.
9. E. Stamatatos. “A survey of modern authorship attribution methods”. In:*Journal of the American Society for information Science and Technology* 60.3 (2009), pp. 538–556