Authorship Verification based on Syntax Features

Jan Rygl, Kristýna Zemková, Vojtěch Kovář

NLP Centre, Faculty of Informatics, Masaryk University

Authorship verification

- Definition:

  1. Confirming or denying authorship by a single known author. [1, 2004]

  2. Given a set of documents written by a suspect along with a document dataset collected from the sample population, we want to determine whether or not an anonymous document is written by the suspect. [2, 2010]

Authorship verification

- Algorithms:
  1. A simple machine learning approach:
     - Extract normalized features from documents D1 (f^1_1, f^1_2, ...) and D2 (f^2_1, f^2_2, …)
     - Count absolute differences of features (similarity): D1 ~ D2 = (1-|f^1_1-f^2_1|, 1-|f^1_2-f^2_2|, …)
     - Train a machine learning classifier using the similarity vector

Authorship verification

- Algorithms:
  2. A ML approach utilizing sample population:
     - Extract normalized features from unknown documents D1, D2 and from sample documents S1, …, S4
     - Count absolute differences of features for:

       D1 ~ D2, D1 ~ S1, D1 ~ S2, D1 ~ S3, D1 ~ S4
     - Compute “Ranking score vector” (R):
       - Ranking of a document X is number of documents more similar to D1 than X according to the feature I.
       - R[i] = 1/(1+ranking of D2 according to feature i)
     - Train a machine learning classifier using the R

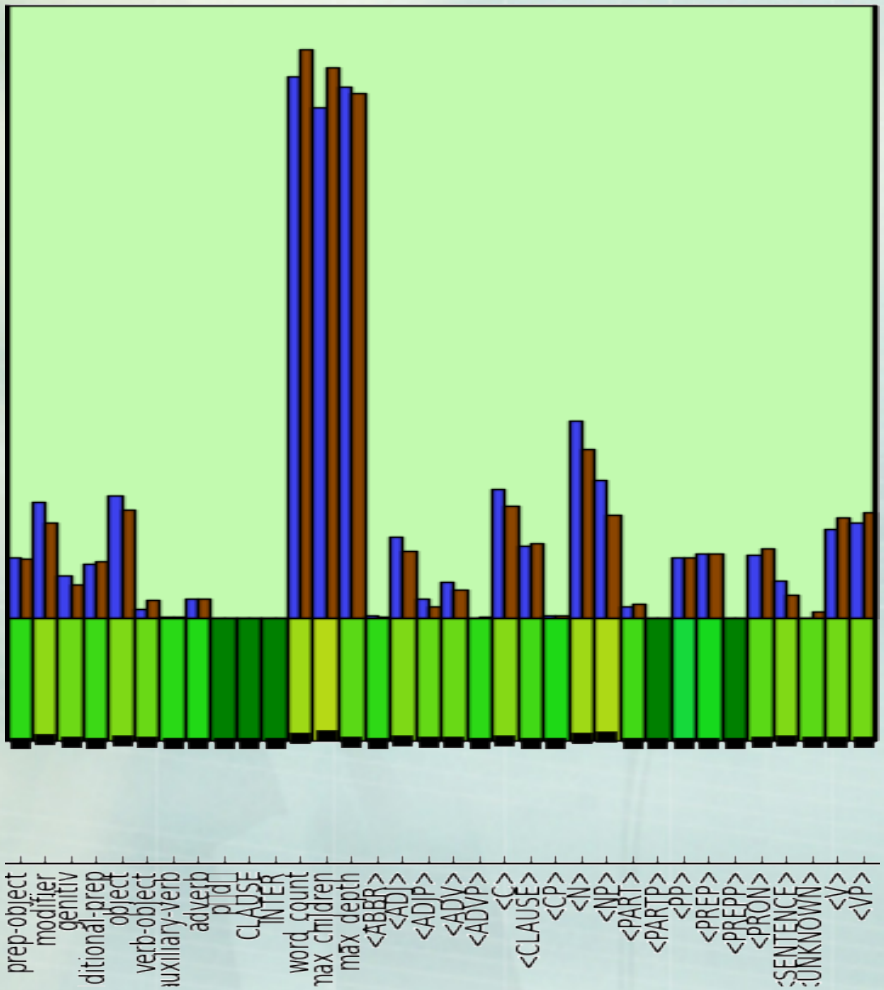
Syntactic analysis using SET

- SET[3] produces parsing trees in three possible output fomats:
  - dependency format (-d option),
  - constituent format (-p option)
  - and hybrid format (default)

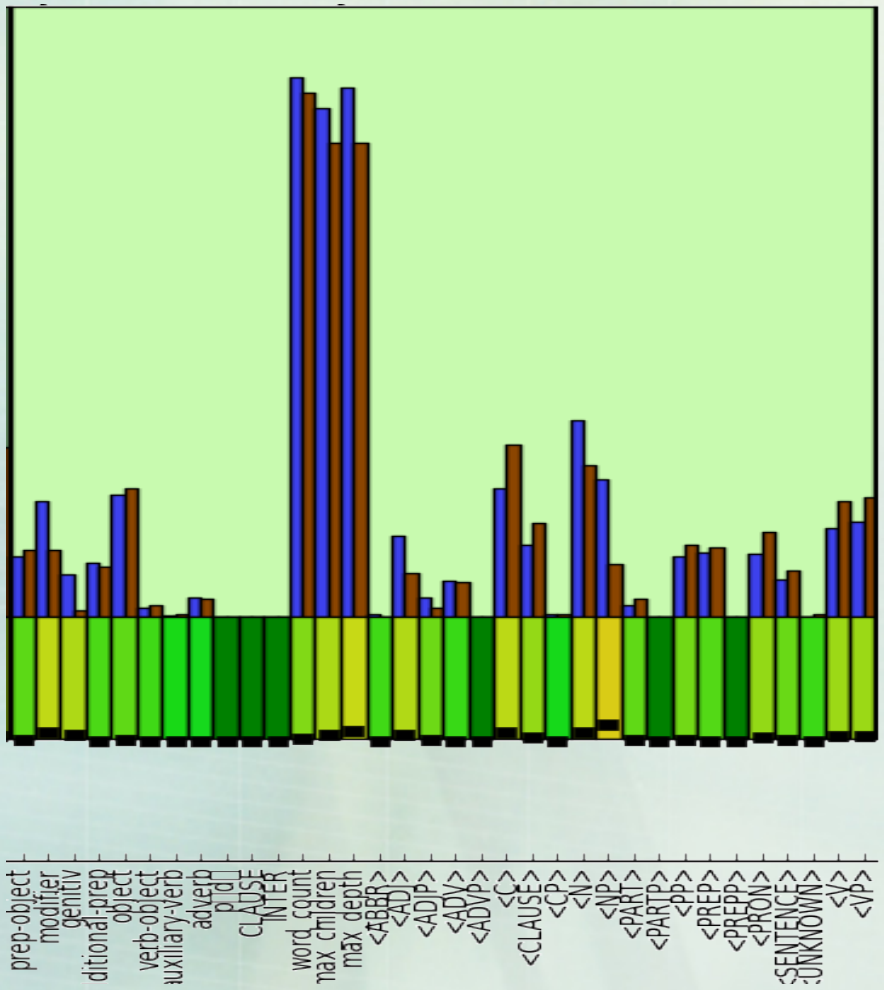
- Selected features from dependency and constituent formats:
  - maximum depth of the dependency tree
  - highest number of child nodes in the dependency tree
  - absolute and relative frequencies of particular non-terminals in the phrasal tree (e.g. <CLAUSE>, <NP>, <VP>)
  - absolute and relative frequencies of particular dependency labels in the dependency tree (e.g. prep-object, verb-object)

Visualization

- Same authors



- Different authors



Results

The simple approach: avg. accuracy 57.9 %

(a) Folder 1: Accuracy: 51.1 %

	Positive	Negative
True	280 (38.5 %)	92 (12.6 %)
False	272 (37.4 %)	84 (11.5 %)

(b) Folder 2: Accuracy: 55.4 %

	Positive	Negative
True	360 (41.7 %)	119 (13.8 %)
False	313 (36.2 %)	72 (8.3 %)

(c) Folder 3: Accuracy: 67.7 %

	Positive	Negative
True	230 (33.6 %)	233 (34.1 %)
False	109 (15.9 %)	112 (16.4 %)

(d) Folder 4: Accuracy: 57.2 %

	Positive	Negative
True	224 (28.7 %)	222 (28.5 %)
False	168 (21.5 %)	166 (21.3 %)

```
Folder 1: Train accuracy 77.4 % for parameters c=2.0 g=0.5
Folder 2: Train accuracy 75.5 % for parameters c=8.0 g=0.5
Folder 3: Train accuracy 70.2 % for parameters c=2048.0 g=0.125
Folder 4: Train accuracy 73.3 % for parameters c=2048.0 g=0.125
```

The ranking approach: avg. accuracy 71.3 %

(a) Folder 1: Accuracy: 79.3 %

	Positive	Negative
True	691 (34.6 %)	894 (44.7 %)
False	106 (5.3 %)	309 (15.4 %)

(b) Folder 2: Accuracy: 64.3 %

	Positive	Negative
True	364 (18.2 %)	921 (46.0 %)
False	79 (4.0 %)	636 (31.8 %)

(c) Folder 3: Accuracy: 69.0 %

	Positive	Negative
True	481 (24.1 %)	899 (44.9 %)
False	101 (5.1 %)	519 (25.9 %)

(d) Folder 4: Accuracy: 72.8 %

	Positive	Negative
True	491 (24.6 %)	965 (48.2 %)
False	35 (1.8 %)	509 (25.4 %)

```
Folder 1: Train accuracy 88.9 % for parameters c=512.0 g=0.125
Folder 2: Train accuracy 88.2 % for parameters c=2048.0 g=2.0
Folder 3: Train accuracy 88.0 % for parameters c=8.0 g=2.0
Folder 4: Train accuracy 87.7 % for parameters c=8.0 g=2.0
```

The simple approach for Word-Length features: avg. accuracy 53.2 %

The ranking approach for Word-Length features: avg. accuracy 61.5 %

Literature

- [1] Hans van Halteren. Linguistic profiling for author recognition and verification. In Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, ACL ’04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.
- [2] Farkhund Iqbal, Liaquat A. Khan, Benjamin C. M. Fung, and Mourad Debbabi. e-mail authorship verification for forensic investigation. In Proceedings of the 2010 ACM Symposium on Applied Computing, SAC ’10, pages 1591–1598, New York, NY, USA, 2010. ACM.
- [3] Vojtěch Kovář, Aleš Horák, and Miloš Jakubíček. Syntactic analysis using finite patterns: A new parsing system for czech. In Zygmunt Vetulani, editor, LTC, volume 6562 of Lecture Notes in Computer Science, pages 161–171. Springer, 2009.