Developing a corpus of plagiarized short answers [Clough and Stevenson, 2011]

Björn Rudzewitz[1]
University of Tübingen

Hauptseminar Language Variation and Stylometrics
WS 15/16

December 16, 2015

[1]brzdwtz@sfs.uni-tuebingen.de

To avoid the objection of plagiarism:

ideas and examples in this presentation are taken from Clough and Stevenson [2011]

Motivation

- correlation between availability of electronic resources and plagiarism
- plagiarism detection as a field suffering from lack of standardized evaluation resources
- previous corpus creation efforts suboptimal:
  - lack of **data** ('deception', how to find plagiarized text)
  - lack of **gold labels** (authors deny judgments)
  - lack of **legal and ethical basis** for data publication
  - lack of **transparency** in data preparation
    ($\rightarrow$ *Leech's maximes* for corpus creation)

Impact and application

Desired effects of the corpus:

- new resource for comparative evaluation and pedagogical methods
- enable new work on plagiarism detection and task strategies

Related work

- Microsoft Research Paraphrase Corpus [Dolan et al., 2004]
- Multiple-Translation Chinese Corpus [Pang et al., 2003]
- METER corpus [Gaizauskas et al., 2001]
- Corpus for plagiarism detection [Zu Eissen et al., 2007]
- PAN Plagiarism detection corpus [Eiselt and Rosso, 2009]

More related resources in Machine Translation evaluation and Short Answer Assessment.

High-level perspective on approaches

- *extrinsic*
  - comparison of source and (potentially) plagiarized text
  - authorship attribution approaches
- *intrinsic*
  - comparison of text passages in one document with each other
  - stylometric approaches

Problem: documents can plagiarize $n \in \mathbb{N}_0$ other documents in different ways

$\rightarrow$ interaction between extrinsic and intrinsic analysis desirable

Plagiarism Techniques: How to plagiarize

Goal: produce an answer of 200-300 words to a question

- *Near copy*
  - copy-and-paste (parts of) Wikipedia article

Plagiarism Techniques: How to plagiarize

Goal: produce an answer of 200-300 words to a question

- *Near copy*
  - copy-and-paste (parts of) Wikipedia article
- *Light revision*
  - like light revision, but with possibility to replace words with synonyms, (lexical/morphosyntactic) paraphrasing
  - information structure preserved

Plagiarism Techniques: How to plagiarize

Goal: produce an answer of 200-300 words to a question

- *Near copy*
  - copy-and-paste (parts of) Wikipedia article
- *Light revision*
  - like light revision, but with possibility to replace words with synonyms, (lexical/morphosyntactic) paraphrasing
  - information structure preserved
- *Heavy revision*
  - rephrasing/paraphrasing of Wikipedia article, n-to-m sentence alignment

Plagiarism Techniques: How to plagiarize

Goal: produce an answer of 200-300 words to a question

- *Near copy*
  - copy-and-paste (parts of) Wikipedia article
- *Light revision*
  - like light revision, but with possibility to replace words with synonyms, (lexical/morphosyntactic) paraphrasing
  - information structure preserved
- *Heavy revision*
  - rephrasing/paraphrasing of Wikipedia article, n-to-m sentence alignment
- *Non-plagiarism*
  - no access to Wikipedia
  - participants read material, then answer question with their (partly freshly) acquired knowledge

Corpus Creation

- 19 participants, CS students
- each participant writing answer for each task (2 times non-plagiarism)
  $\rightarrow$ 95 answers + 5 articles = 100 documents (19,995 tokens)
- Graeco-Latin Square Design for systematic randomization and rotation of revision types per participant and question
- participant meta data: native language, familiarity with answer, perceived difficulty of task

$\mu_{tok/aw} = 208$ $\sigma_{tok/aw} = 64.91$
$\mu_{types/aw} = 113$ $\sigma_{types/aw} = 30.11$

Data Analysis: Individual Differences

- statistically significant difference ($p < 0.01$) between native and non-native speakers wrt. mean knowledge and perceived difficulty (two-sample t-test)

  $\rightarrow$ difference in population means of two independent samples
- Positive Pearson's correlation of $r = 0.344$ between knowledge and perceived difficulty

Data Analysis: Observations

Question:

A. What is inheritance in object oriented programming?

Example 1:

Inheritance allowes classes to be categorized, similer to the way humans catagorize. It also provides a way to generalize du to the "is a" relationship between classes.

Example 2:

Generlisation also some time known as inheritance. The main reason behind this is a hierarchi st ructure of objects and classes. We can understand this mechanism by some examples: like fruit i s aq main class and mangoes apple ,orange is child classs of the main class.So obviously inherit all the properties of fruit class.

Data Analysis: Observations

Question:

A. What is inheritance in object oriented programming?

Example 1:

Inheritance allowes classes to be categorized, similer to the way humans catagorize. It also provides a way to generalize du to the "is a" relationship between classes.

Example 2:

Generlisation also some time known as inheritance. The main reason behind this is a hierarchi st ructure of objects and classes. We can understand this mechanism by some examples: like fruit i s aq main class and mangoes apple ,orange is child classs of the main class.So obviously inherit all the properties of fruit class.

Data Analysis: Observations

Question:

A. What is inheritance in object oriented programming?

Example 1:

Inheritance allowes classes to be categorized, similer to the way humans catagorize. It also provides a way to generalize du to the "is a" relationship between classes.

Example 2:

Generlisation also some time known as inheritance. The main reason behind this is a hierarchi st ructure of objects and classes. We can understand this mechanism by some examples: like fruit i s aq main class and mangoes apple ,orange is child classs of the main class.So obviously inherit all the properties of fruit class.

spelling mistake

missing predicate

missing subject

segmentation mistake

Data Analysis: Observations

Question:

A. What is inheritance in object oriented programming?

Example 1:

Inheritance allowes classes to be categorized, similer to the way humans catagorize. It also provides a way to generalize du to the "is a" relationship between classes.

Example 2:

Generlisation also some time known as inheritance. The main reason behind this is a hierarchi st ructure of objects and classes. We can understand this mechanism by some examples: like fruit i s aq main class and mangoes apple ,orange is child classs of the main class.So obviously inherit all the properties of fruit class.

spelling mistake

missing predicate

missing subject

segmentation mistake

} need for robust processing resources

Experimental Automatic Plagiarism Detection

2 classification tasks:

1. Prediction of plagiarism and plagiarism type:
   Predict a class c with

$$c \in \{\text{"near copy"}, \text{"light revision"}, \text{"heavy revision"}, \text{"non-plagiarism"}\}$$

2. Binary classification of plagiarism:
   Predict a class c with

$$c \in \{\text{"plagiarism"}, \text{"non-plagiarism"}\}$$

2 feature types: n-gram overlap, LCS

N-Gram Overlap

n-gram containment on document level

$$c_n(A, B) = \frac{|S(A, n) \cap S(B, n)|}{|S(A, n)|}$$

$n \in \mathbb{N}, 0 < n < 6$ (window size)
A, B documents

Longest Common Subsequence (LCS)

- longest shared (possibly) non-continuous sequence
- compute minimum number of edit operations for transforming text A into B
- normalized lcs: normalize by length of answer text

LCS

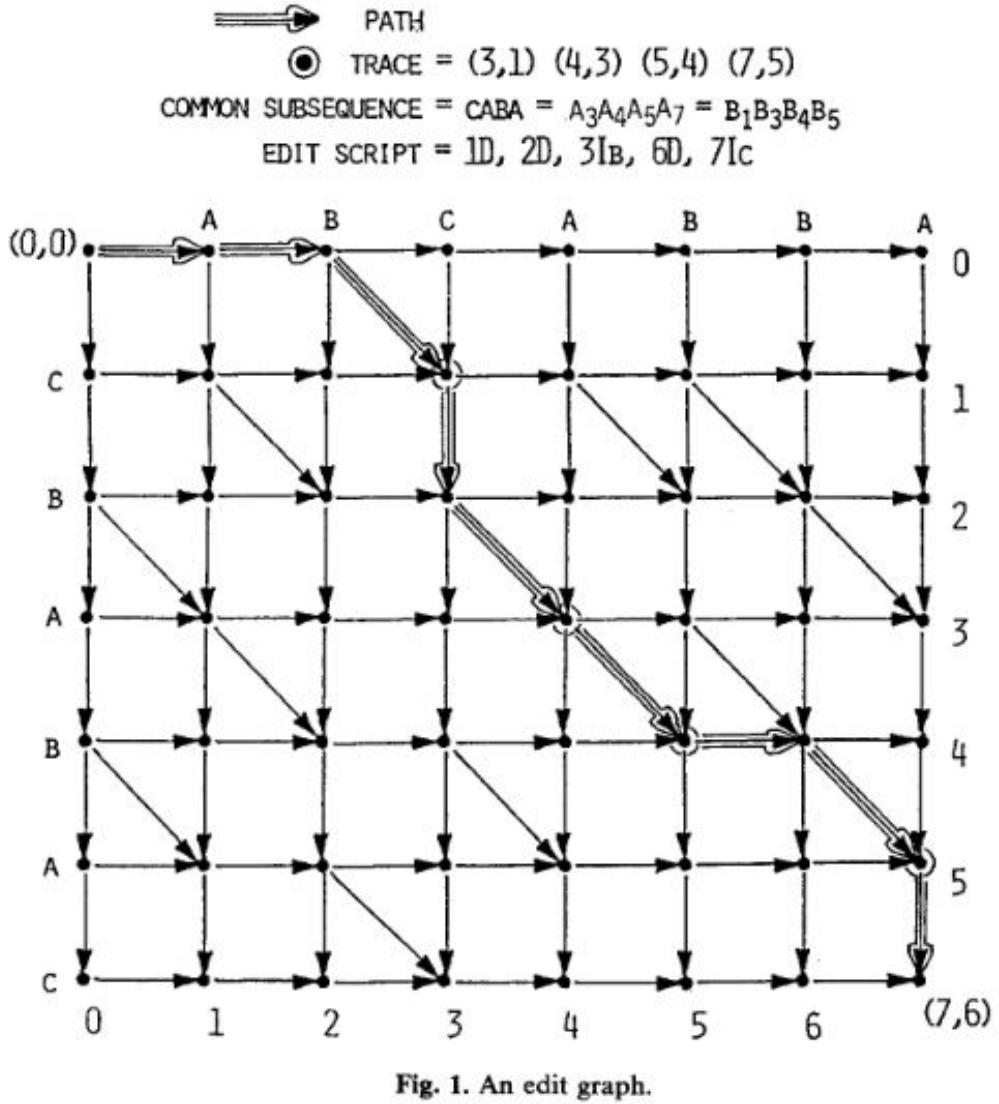


Fig. 1. An edit graph.

Figure: Relation between Longest Common Subsequence and Edit Operations (from [Myers, 1986, page 253])

LCS

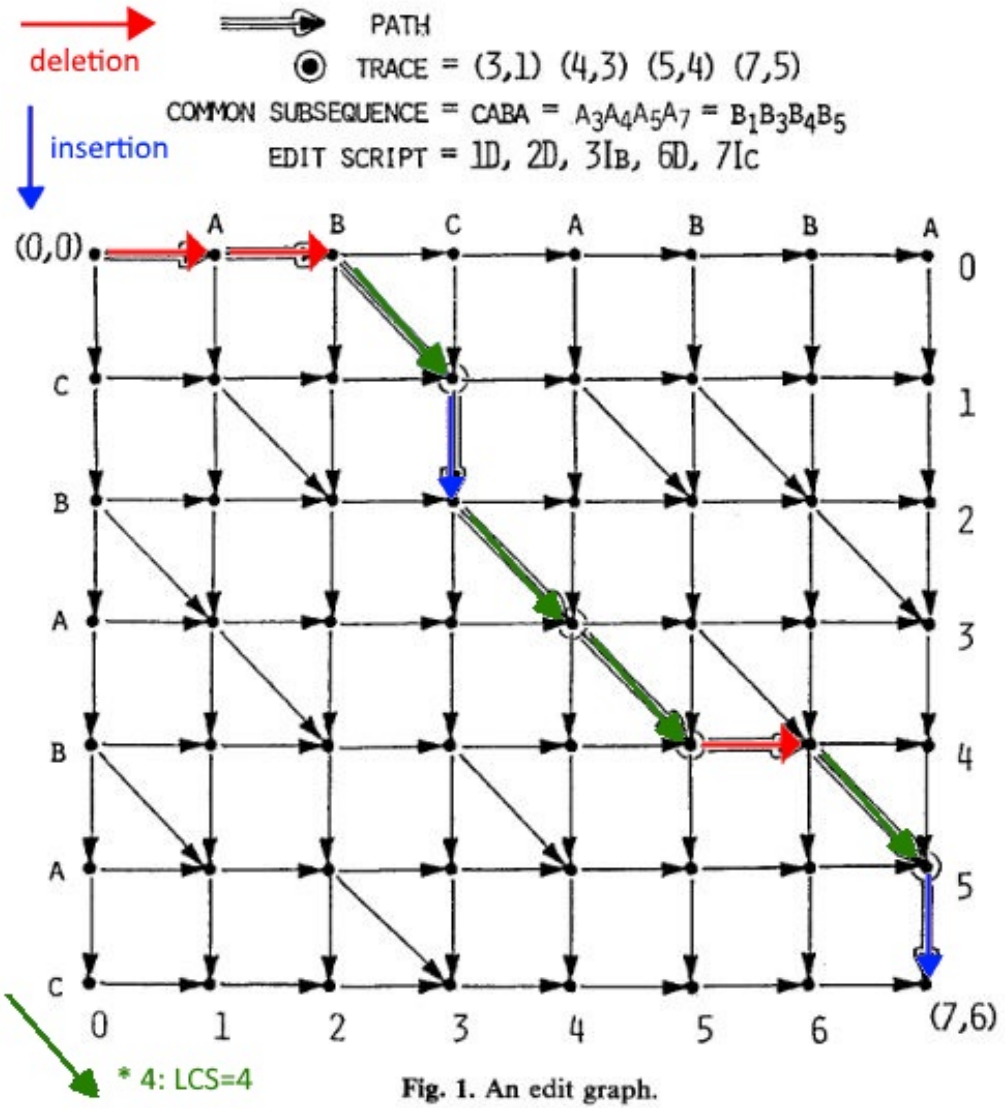


Fig. 1. An edit graph.

Figure: Relation between Longest Common Subsequence and Edit Operations (from [Myers, 1986, page 253])

Baselines

Comparison of answers with unrelated articles

Task	$c_w(A,B)$ for w-gram					lcs_{norm}
	1	2	3	4	5	
A	0.48	0.15	0.08	0.05	0.03	0.26
B	0.65	0.23	0.12	0.08	0.05	0.35
C	0.49	0.20	0.11	0.06	0.03	0.29
D	0.60	0.29	0.17	0.10	0.06	0.35
E	0.61	0.23	0.13	0.08	0.05	0.34
Avg.	0.57	0.22	0.12	0.07	0.04	0.32

- high unigram overlap between topic-unrelated answers and Wikipedia articles

Baselines

Comparison of answers with related articles

Category	$c_n(A,B)$ for n-gram					lcs_{norm}
	1	2	3	4	5	
Near copy	0.95	0.89	0.85	0.81	0.78	0.88
Light revision	0.87	0.70	0.56	0.46	0.39	0.76
Heavy revision	0.81	0.52	0.34	0.26	0.21	0.58
Non-plagiarised	0.63	0.23	0.05	0.01	0.00	0.41

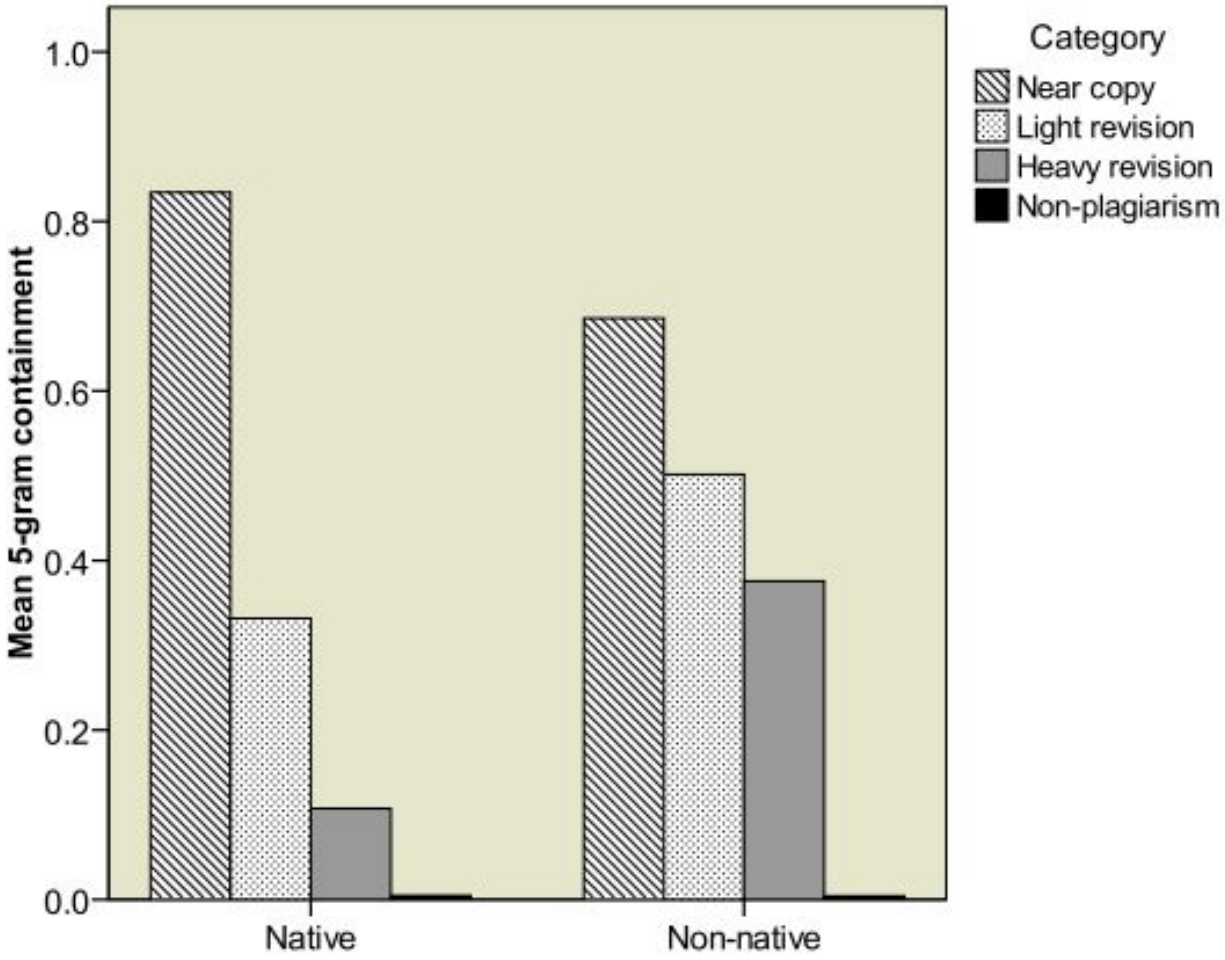
- high n-gram overlap between topic-related answers and Wikipedia articles
- less strong drop for higher n
- statistically significant differences between similarity of rewrite levels with articles (ANOVA with Bonferroni pos-hoc test)

Comparison of answers by question ('task')

Task	$c_n(A,B)$ for n-gram					lcs_{norm}
	1	2	3	4	5	
A	0.77	0.45	0.31	0.27	0.25	0.55
B	0.81	0.53	0.35	0.28	0.25	0.63
C	0.71	0.44	0.31	0.25	0.21	0.53
D	0.82	0.58	0.46	0.40	0.36	0.69
E	0.81	0.56	0.41	0.35	0.32	0.65
Avg	0.79	0.51	0.37	0.31	0.28	0.61

- averaging over all (non)plagiarism types
- 'most' differences not significant

L1 vs L2



- higher n-gram containment scores for non-natives for heavier revision
- insignificant, though noticeably higher amount of lifting of material for participants writing in L2

Classification

Class	Feature						
	$c_1(A,B)$	$c_2(A,B)$	$c_3(A,B)$	$c_4(A,B)$	$c_5(A,B)$	lcs_{norm}	all
Near copy	0.778	0.778	0.850	0.850	0.829	0.571	0.850
Light revision	0.605	0.579	0.571	0.452	0.357	0.400	0.629
Heavy revision	0.457	0.485	0.500	0.500	0.537	0.556	0.611
Non-plagiarised	0.895	0.937	0.911	0.902	0.925	0.911	0.937
Overall accuracy	72.6%	76.8%	75.8%	73.7%	73.7%	67.4%	80.0%

- Naive Bayes Classifier from WEKA
- best result for binary classification: 94.3% accuracy
- best result for classification of 4 classes: 80.0% accuracy

$$F_1 = 2 * \frac{precision * recall}{precision + recall}$$

Conclusion

- publicly available corpus of manually created plagiarized text: *Wikipedia Reuse Corpus*
- different types of plagiarism represented, authentic language
- simple features allowed plagiarism classification with 95% accuracy

Discussion

- Text of 200-300 words usually not considered a short answer in SAA (e.g. Burrows et al. [2015], Ziai et al. [2012])
- Probability of academics copying verbatim from Wikipedia ?
- Are students working on a plagiarism project representative of the population of participants ?
- Are measures of central tendency for very heterogenous data justified ?
- "questionable gold standard annotation" [Zesch and Gurevych, 2012, page 174] ?

Steven Burrows, Iryna Gurevych, and Benno Stein. The eras and trends of automatic short answer grading. *International Journal of Artificial Intelligence in Education*, 25(1):60–117, 2015.

Paul Clough and Mark Stevenson. Developing a corpus of plagiarised short answers. *Language Resources and Evaluation*, 45(1):5–24, 2011.

Bill Dolan, Chris Quirk, and Chris Brockett. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th international conference on Computational Linguistics*, page 350. Association for Computational Linguistics, 2004.

Martin Potthast Benno Stein Andreas Eiselt and Alberto Barrón-Cedeno Paolo Rosso. Overview of the 1st international competition on plagiarism detection. In *3rd PAN Workshop. Uncovering Plagiarism, Authorship and Social Software Misuse*, page 1, 2009.

Robert Gaizauskas, Jonathan Foster, Yorick Wilks, John Arundel, Paul Clough, and Scott Piao. The meter corpus:

a corpus for analysing journalistic text reuse. In *Proceedings of the Corpus Linguistics 2001 Conference*, pages 214–223, 2001.

Diane Kelly. Methods for evaluating interactive information retrieval systems with users. *Foundations and Trends in Information Retrieval*, 3(1-2):1–224, 2009.

Eugene W Myers. An o(nd) difference algorithm and its variations. *Algorithmica*, 1(1-4):251–266, 1986.

Bo Pang, Kevin Knight, and Daniel Marcu. Syntax-based alignment of multiple translations: Extracting paraphrases and generating new sentences. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 102–109. Association for Computational Linguistics, 2003.

Daniel Bär1 Torsten Zesch and Iryna Gurevych. Text reuse detection using a composition of text similarity measures. In *Proceedings of COLING*, volume 1, pages 167–184, 2012.

Ramon Ziai, Niels Ott, and Detmar Meurers. Short answer assessment: Establishing links between research strands. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 190–200. Association for Computational Linguistics, 2012.

Sven Meyer Zu Eissen, Benno Stein, and Marion Kulig. Plagiarism detection without reference collections. In *Advances in data analysis*, pages 359–366. Springer, 2007.