Style & Identity Recognition

Jan Rygl

Natural Language Processing Centre,
Faculty of Informatics, Masaryk University
Botanická 68a, 602 00 Brno, Czech Republic
rygl@fi.muni.cz

Abstract. Knowledge of the author's identity and style can by used in the fight against forged and and anonymous documents and illegal actions in the Internet. Nowadays, there are many systems dedicated to solving stylometric tasks, but they are predominantly designed only for a specific task; they are used exclusively by their owners; or they do not natively support any Slavic languages.
Therefore, we present new open-source modular system *Style & Identity Recognition* (*SIR*). The system is designed to support any stylometric tasks with minimal efforts (or event by default) by combining dynamic stylometry features selection and prediction driven by input data labels. The system is free for non-commercial applications and easy to use, therefore it can be helpful for people dealing with threatening e-mails or sms, children forum protection against pedophiles and other tasks. Being customizable and freely accessible, it can be also used as a baseline for other systems solving stylometry tasks.
System combines machine learning techniques and nature language processing tools. It is written in Python and it is dependent on other open-source Python libraries.

Keywords: stylometry, authorship recognition, machine learning, open-source

1 Introduction

Organizations such as forensic expert bureaus, national security agencies and big companies are using advanced language tools to predict style and identity of author of the text.

But the tools which these organizations are using are predominantly limited by two factors:

1. **Exclusive rights**
   e.g. *ART*[1] [1] is used by Ministry of the Interior of CR,
   or *FLAIR*[2] is used only by a forensic expert bureau.

[1] *ART*: Authorship Recognition Tool

[2] *FLAIR*: Forensic Linguistic Advice, Investigation and Research, see
http://www.forensiclinguistics.net/

Aleš Horák, Pavel Rychlý, Adam Rambousek (Eds.): Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN 2015, pp. 3–10, 2015.

2. **No support for other languages such as Slavic**
   e.g. *JGAAP*[3][2] lastly updated in 2013.

The main usage of the stylometry tools includes the following tasks:

1. authorship verification (court evidence; Internet authentication)
2. authorship attribution (criminal investigation of anonymous illegal documents)
3. authorship clustering (multiple authorship detection in essays, multiple accounts illegally used by one user [3])
4. age prediction (pedophile detection in children web communication [4])
5. translation detection (was the text translated by a person or by an automatic method?)
6. mental illness recognizer (detect symptoms and warn people)
7. personality analyzer (predict personality traits for human resources)
8. ...

Most of existing tools are specialized and not publicly available. Therefore, there is a place for an accessible free tool which can handle any stylometry task and can be used by non-expert users with almost optimal performance. Our goal is to fill this gap, therefore we present *Style & Identity Recognition* (*SIR*) tool.

2 Stylometry analysis

Author's style can be defined as a set of measurable text features (style markers) according to stylostatisticians [5]. Definition can be extended by adding non-text features such as colors, link domains and publication times.

The good example of style markers are frequencies of word-lengths. They were used as the first deterministic stylometry technique to detect an authorship of documents. T. C. Mendenhall discovered that word-length frequency distribution tends to be consistent for one author and differs for different authors (1887, [6]).

Style markers can depend on the properties of texts (formatting richness) and by tools which were used to extract them.

Modern methods use machine learning to process style markers extracted from documents. Machine learning techniques such as Support Vector Machines [7] and Random Forests always outperform pure distance metrics such as cosinus similarity (used for example in authorship verification).

3 Components of style & identity recognition

1. Stylometry corpus builder
2. Text cleaning (boiler-plate removal, HTML removal, etc.)

[3] *JGAAP*: Java Graphical Authorship Attribution Program, see `www.jgaap.com`

3. Language detection
4. Encoding detection
5. Text tokenization and further analysis
6. Semantic analysis (entity detection, abbrevation expansion, etc.)
7. Style markers selection
8. Style markers extraction
9. Machine learning processing

3.1 Stylometry corpus builder

Since we are using machine learning techniques, we need documents to tune features (style markers extractors), to train classifiers and to evaluate them. For English and other majority languages, there are many available language sources (e.g. e-mail corpus Enron [8]).

But for Slavic languages such as Czech and Slovak, there are several very small manually collected collections (e.g. Czech essays of pupils [9]). But there is also a current project *Authorship corpora builder* (*ACB*)[10] focused on small European languages. *ACB* contains free pre-built corpora for Czech and Slovak languages and tools for building new corpora. The tool and built corpora are freely accessible at `https://nlp.fi.muni.cz/projekty/acb/`.

Since there are existing tools and data sources, data collection is not planned to be part of *SIR* tool.

3.2 Text cleaning

Boiler-plate and markup removal phase is the most effective if it is done during the process of data crawling (we know the structure of the data domain and can compare an analyzed document with a big set of documents – even with documents not meant to pass data selection process).

Therefore, we use text cleaning already present in data sources and do not perform further text cleaning in *SIR* tool.

3.3 Language detection

Language detection is very important because style markers (machine learning features) depend on the language of documents. Features based on morphology, syntactic analysis or entity detection require to be given the language of a document before a document procession step.

Our system uses *langid* [11] library (`https://github.com/saffsd/langid.py`).

3.4 Encoding detection

Despite the fact that more than 85% of web pages use `utf-8` encoding [12], the encoding detection process can be still useful. Middle-European languages

such as Czech and Slovak use non-ascii characters and bad encoding detection can negatively influence text post-processing (e.g. morphology analysis). We recommend to use Chared[4] library, our *SIR* tool natively supports it.

3.5 Text tokenization and further analysis

There are many libraries supporting naive text tokenization (word separation based on white spaces and punctuation). In language independent application, we need one robust general tokenizer usable for all languages. If we have a specialized tokenizer for given languages, it can be used instead of a general one.

As a general tokenizer, `nltk.tokenize.WordPunctTokenizer` is used. The *SIR* tool also supports morphology analysis using RFTagger [13]. If RFTagger is used, not only morphology tagging is performed for supported languages (Czech, Slovak, Slovene, German, Hungarian, Russian), but also tokenization is done by RFTagger.

In following versions, support for other morphology taggers and syntactic analyzers will be added. The taggers are not part of the project and they are used as external libraries instead because of licensing restrictions.

3.6 Semantic analysis

There are two reasons not to implement general multilingual semantic analysis:

1. For each language, we need implementation of one semantic analyzer (e.g. named entity in one language is not a named entity in another language).
2. Style markers usually use only small part of semantic analysis output, therefore it is better to make specialized standalone analysis for each style-markers extraction (which can be faster and more accurate than complex analysis).

We decided that semantic analysis should be part of style-markers extraction phase.

3.7 Style markers selection

Style markers are divided into two categories:

1. Language independent style markers (e.g. word length, sentence length, capitalization)
2. Language dependent (e.g. syntactic analysis)

Special case is a morphology analysis. We are using RFTagger which uses similar tagset for all supported languages, therefore some style markers can be language dependent, but support wide range of languages.

[4] `http://nlp.fi.muni.cz/projects/chared/`

The quality and the utility of style markers depend on the type of solved problem. Different document lengths and tasks require different style markers, therefore it is recommended to experimentally select a subset of style markers and not to use them all [14].

Style marker selection is a semi-automatic step, only style markers supporting language of a document are preselected, but user can narrow the selection by filtering out categories of style markers not suitable for current problem.

3.8 Style markers extraction

For each category of style markers (e.g. word length is a category, `word length 1`, `word length 2`, ...are style markers), there is one python class. *SIR* tool includes several implementations of established style-markers categories and others will be added in future. Users are allowed to add new categories depending on their demand, each category is defined by:

- feature list (e.g. stopwords `game`, `tv`, `chat`, `facebook`)
- for each feature, a function converting processed document (text, morphology analysis, title, publication time) to a float number (e.g. if game in text: features[0] = "game" in text).

3.9 Machine learning processing

We use *scikit-learn* [15] library. Default machine learning algorithm is Random Forest Classifier, but in future versions we will support automatic classifier selection.

Classifier parameters are found using cross-validation on train data and native *scikit-learn* grid search.

All features are scaled to range $\langle 0, 1 \rangle$.

4 SIR

The project is developed under *Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License*[5].

The system is implemented in Python and uses following third party libraries: *gensim*, *ipython*, *numpy*, *requests*, *scikit-learn*, *scipy*, *smart-open*, *sqlitedict*, *xmltodict*, *langid*, *Flask*, *Flask-Cors*, *Flask-Mako*, *gunicorn*, *argparse*, *nltk*, and *chared*. It also supports morphological analyzer RFTagger [13].

The project is located on *GitHub* at url `https://github.com/janrygl/sir`. Online demo is available at `nlp.fi.muni.cz/sir`.

[5] `https://creativecommons.org/licenses/by-nc-nd/4.0/`

5 Evaluation

For evaluation purposes, we used reference corpus 2.0 of *Authorship corpora builder*[6]. The **authorship attribution** problem was solved: *Given a particular sample of text known to be by one of a set of authors, determine which one* [2, p. 238].

We used Czech documents from the reference corpus and tested scenarios with 2, 5, 10, 15, 20 and 28 authors. To be objective, we ran tests for each candidate count 100times (except the highest 28 authors), each time randomly selecting different authors. Resulting accuracies and standard deviations are displayed in Table 1 and in Figure 1.

Table 1. Authorship attribution experiment.

Author count	Accuracy	Baseline	Iterations
2	84% ± 16%	50.00%	100 iterations
5	60% ± 15%	20.00%	100 iterations
10	49% ± 10%	10.00%	100 iterations
15	42% ± 8%	6.67%	100 iterations
20	39% ± 6%	5.00%	100 iterations
28	41% ± 0%	3.57%	1 iterations

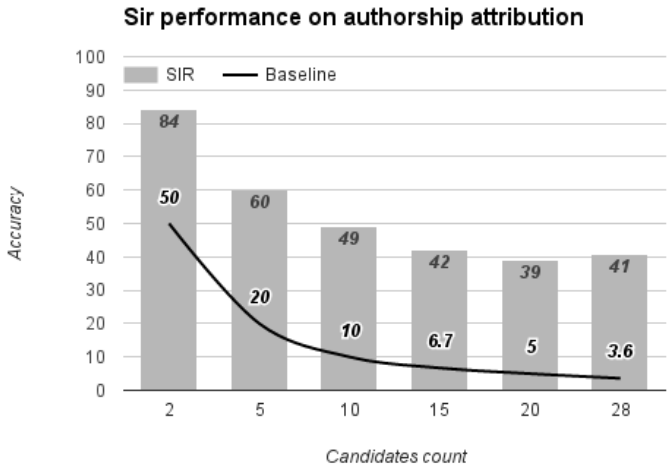

Fig. 1. Authorship attribution: results vs baseline.

[6] `https://nlp.fi.muni.cz/projekty/acb/getfile?name=download/author_corpus_v2.zip`

With growing number of candidates, accuracy is decreasing, but experimental results indicate that achieved accuracies are reasonably high and stable (reasonable standard deviation).

6 Conclusions and future work

We have introduced universal stylometric system ready to analyze documents. System can be downloaded from `https://github.com/janrygl/sir`.

We are going to actively develop system and add new features. Our plan is to provide accessible system that can by used by common Internet users to help them solve their stylometric tasks such as authorship attribution and gender recognition.

Acknowledgements This work has been partly supported by the Ministry of Education of CR within the LINDAT-Clarin project LM2010013.

References

1. Rygl, J.: Art: Authorship recognition tool (2014)
2. Joula, P.: Authorship Attribution. Foundations and Trends in Information Retrieval. (2008)
3. Verhoeven, B., Daelemans, W.: Clips stylometry investigation (csi) corpus: A dutch corpus for the detection of age, gender, personality, sentiment and deception in text. In Calzolari, N., Choukri, K., Declerck, T., Loftsson, H., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., Piperidis, S., eds.: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland, European Language Resources Association (ELRA) (May 2014) 3081–3085 ACL Anthology Identifier: L14-1001.
4. Peersman, C., Daelemans, W., Van Vaerenbergh, L.: Predicting age and gender in online social networks. In: Proceedings of the 3rd International Workshop on Search and Mining. SMUC '11, New York, NY, USA, ACM (2011) 37–44
5. Stamatatos, E., Fakotakis, N., Kokkinakis, G.: Automatic text categorization in terms of genre and author. Computational Linguistics **26**(4) (Dec 2000) 471–495
6. Mendenhall, T.C.: The characteristic curves of composition. The Popular Science **11** (1887) 237–246
7. Koppel, M., Schler, J., Argamon, S.: Computational methods in authorship attribution. J. Am. Soc. Inf. Sci. Technol. **60** (January 2009) 9–26
8. Klimt, B., Yang, Y.: Introducing the enron corpus. In: CEAS 2004 - First Conference on Email and Anti-Spam, July 30-31, 2004, Mountain View, California, USA. (2004)
9. Šebesta, K., collective of authors from ÚČNK FF UK: SKRIPT2012: akviziční korpus psané češtiny – přepisy písemných prací žáku základních a středních škol v ČR (in English: acquisition corpus of Czech written language – transcripts of the written work of pupils in primary and secondary schools in the Czech Republic) (2013)
10. Švec, J., Rygl, J.: Slavonic corpus for stylometry research. In: Proceedings of Ninth Workshop on Recent Advances in Slavonic Natural Language Processing, RASLAN 2015., Tribun EU, 1st ed. Brno (Czech Republic) (2015)

11. Lui, M., Baldwin, T.: Langid.py: An off-the-shelf language identification tool. In: Proceedings of the ACL 2012 System Demonstrations. ACL '12, Stroudsburg, PA, USA, Association for Computational Linguistics (2012) 25–30
12. W3Techs: http://w3techs.com/technologies/details/en-utf8/all/all
13. Schmid, H., Laws, F.: Estimation of conditional probabilities with decision trees and an application to fine-grained pos tagging. In: Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1. COLING '08, Stroudsburg, PA, USA, Association for Computational Linguistics (2008) 777–784
14. Rygl, J.: Automatic Adaptation of Author's Stylometric Features to Document Types. In: Text, Speech and Dialogue - 17th International Conference. 8655., Brno: Springer, 2014. (2014) 53–61
15. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12** (2011) 2825–2830