CHAPTER 16

Stylene: an Environment for Stylometry and Readability Research for Dutch

Walter Daelemans[a,d], Orphée De Clercq[b] and Véronique Hoste[c]

[a]CLiPS Computational Linguistics Group, University of Antwerp, walter.daelemans@uantwerpen.be, [b]LT3 Language and Translation Technology Team, Ghent University, orphee.declercq@ugent.be, [c]LT3 Language and Translation Technology Team, Ghent University, veronique.hoste@ugent.be,

[d]Corresponding Author: walter.daelemans@uantwerpen.be

ABSTRACT

We describe an educational demonstration interface and tools for stylometry (authorship attribution and profiling) and readability research for Dutch. The Stylene system consists of a popularisation interface for learning about stylometric analysis, and of web-based interfaces to software for readability and stylometry research aimed at researchers from the humanities and social sciences who do not want to develop or install such software themselves.

16.1 Introduction

The last decade has seen a marked increase in research on computational stylometry, the subarea of natural language processing that concerns itself with the categorisation of texts according to the psychological and sociological properties of their authors. Also called text profiling, this research tries to develop systems, mostly based on text analytics techniques, that combine natural language processing and machine learning methods. These systems are trained to determine whether the author of a text is male or female, their education level, region of origin, personality, and even mental health, whether they are a native speaker or not, and many other potentially useful attributes. Of course, authorship attribution research has existed for a long time, and is in a sense the limit case of computational stylometry: supposing that everyone has a unique combination of demographic, psychological and idiosyncratic style properties, this would be their idiolect or 'stylome' (Van Halteren et al, 2005; Coulthard, 2004), and it should be possible to assign texts of unknown authorship to specific authors provided that models of their stylome exist.

How to cite this book chapter:

Daelemans, W, De Clercq, O and Hoste, V. 2017. Stylene: an Environment for Stylometry and Readability Research for Dutch. In: Odijk, J and van Hessen, A. (eds.) *CLARIN in the Low Countries*, Pp. 195–209. London: Ubiquity Press. DOI: https://doi.org/10.5334/bbi.16.

Another useful type of information that can be extracted from text using natural language processing and text categorisation is the readability of a text. Readability research and the automatic prediction of readability has a very long and rich tradition (see surveys by Klare 1976; DuBay 2004; Benjamin 2012; and Collins-Thompson, 2014). Whereas superficial text characteristics leading to on-the-spot readability formulas were popular until the 1990s (Flesch 1948; Gunning 1952; Kincaid et al. 1975), recent advances in the field of computer science and natural language processing have triggered the inclusion of more intricate characteristics in present-day readability research (Si and Callan 2001; Schwarm and Ostendorf 2005; Collins-Thompson and Callan 2005; Heilman et al. 2008; Feng et al. 2010).

Current approaches model lexical, syntactic, semantic and discourse complexity, while also considering shallow traditional text characteristics. Furthermore, the focus has shifted from using the formulas to select reading material for children or L2 language learners to assessing the readability of a variety of text types with other user groups or applications in mind.

This chapter introduces the results of a CLARIN Flanders project on the development of practical tools for stylometry and readability.[1] The goal of that project was to implement a robust, modular system for stylometry and readability research on the basis of existing methods, and the development of a web service that would allow researchers in the humanities and social sciences to analyse texts with this system.

The website has three sub-interfaces: (i) a popularisation interface intended to provide basic insight into what stylometry can do; (ii) a readability interface that allows the input of texts and provides elementary and more advanced feedback on the readability of the text; and (iii) a machine learning interface that allows basic experiments in computational stylometry.

In this chapter, we will describe the underlying methods and approaches in the backend of the interfaces. We also developed a stand-alone system for machinelearningbased stylometry that underlies the third interface, but which allows more options and flexibility as a stand-alone system than can be accessed from the interface. The stand-alone system may eventually replace the corresponding interface on the website.

16.2 The Stylometry Popularisation Interface

Computational Stylometry is not yet well-known outside computational linguistics and the specialised digital humanities research community. In order to educate interested lay persons and humanities and social sciences colleagues about the possibilities (and limitations) of the approach, an interface was designed to help a general audience understand computational stylometry in an easy and fun way. An early version was tested out successfully during the 2011 Flemish 'Wetenschapsweek' (Science Week) with secondary school pupils, and afterwards extended. There has been a large interest for the interface (around 50 visitors per month) and some media attention. Figure 16.1 shows the start screen of the interface. Input can be provided either by cut and paste or through file upload. In both cases the input should be raw text (uploaded files should have .txt extension). The demo will only give complete output in browsers that are HTML5compatible and that allow JavaScript. For practical reasons, cutandpaste input is limited to 4000 characters and file upload to 300 sentences.

After the user enters a text and clicks on 'analyse', the software returns a screen with didactic information about the general approach taken in stylometry and information about different stylometric aspects of the text provided. Figure 16.2 shows the introductory information that is

[1] The interface and backend software we describe here was developed in the context of the Stylene (Stylometrie en Leesbaarheid voor het Nederlands; An environment for stylometry and readability research for Dutch) project. The project was funded by the Flemish Ministry for Economics, Science, and Innovation (EWI). The system was developed over 2010–2012 in a cooperation between the CLiPS and LT3 research groups. The interface can be found at: http://www.stylene.be.

Figure 16.1: Start screen of the stylometry popularisation interface.

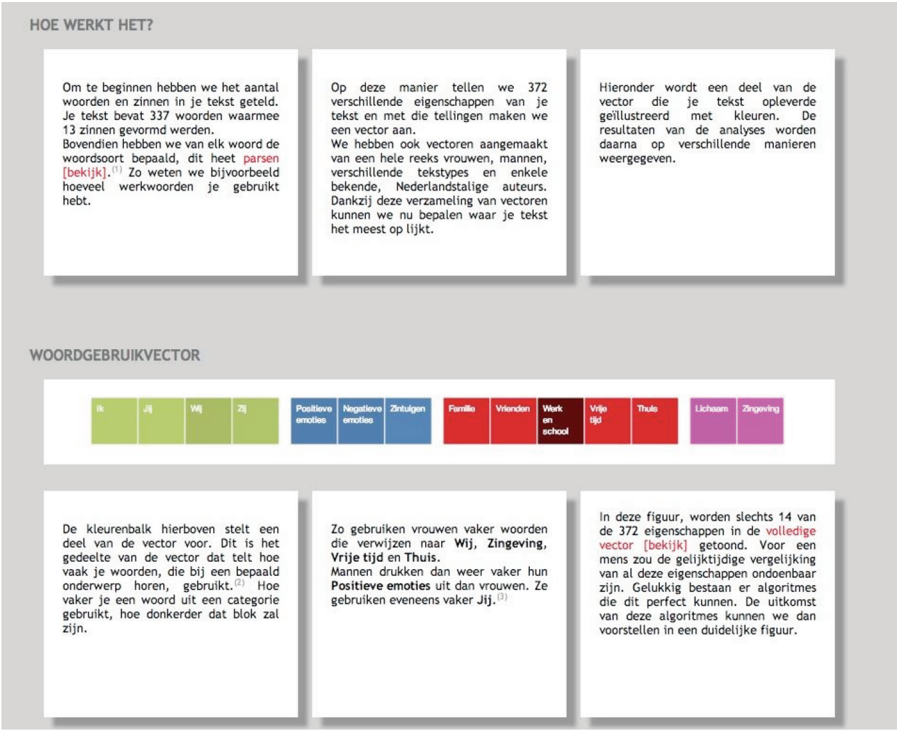

Figure 16.2: Stylometry popularisation interface: output (general).

given about the analysis (users can click through to sample linguistic analyses of the data) and a visual (colour) representation of the distribution of words in the text for some of the features used in the system (the full feature representation can be clicked on as well). Darker features represent more frequent features. For the linguistic analysis, the software package Frog was used (Van den Bosch et al., 2007). As features, token unigrams and the LIWC features (Pennebaker and Francis, 1996; Pennebaker et al 2001; Pennebaker et al, 2007) were used. The latter features group vocabulary associated with specific cognitive and emotional styles and themes, as well as grammatical categories (for example personal pronouns) associated with differences in demographic and psychological properties of authors.

Figures 16.3–16.5 show the additional information that is provided by the demo system: a guess of the gender of the author (based on a model learned by a support vector machine learning algorithm using all features and trained on part of the Corpus Gesproken Nederlands (CGN 2004)

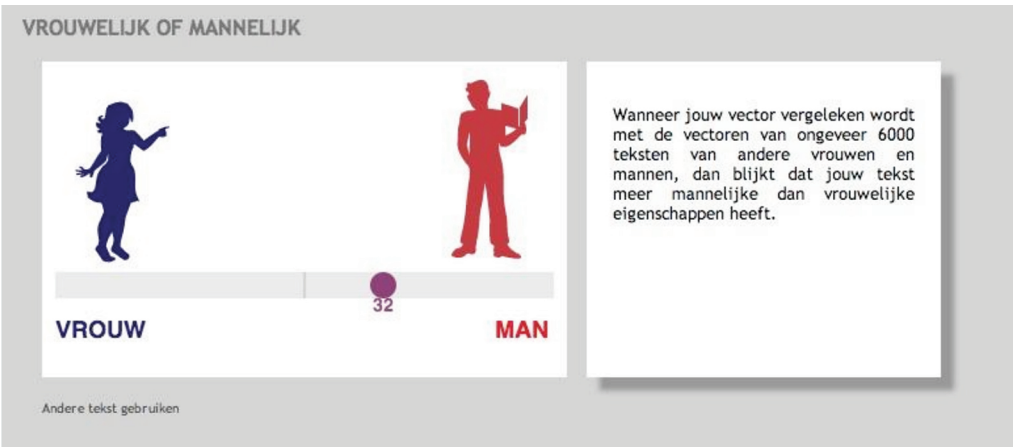

Figure 16.3: Stylometry popularisation interface: output (gender).

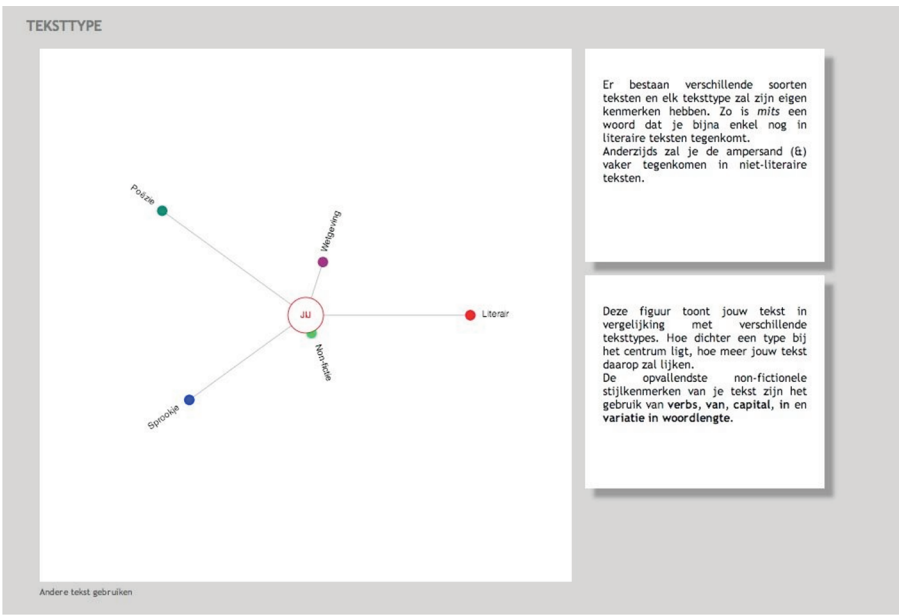

Figure 16.4: Stylometry popularisation interface: output (genre).

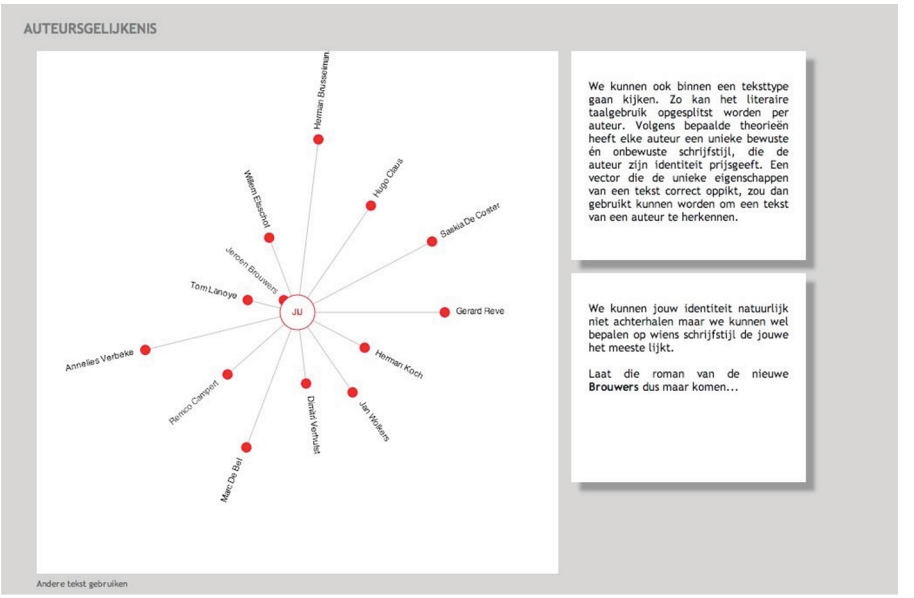

Figure 16.5: Stylometry popularisation interface: output (distance to authors).

data; Figure 16.3); a guess of the genre of the text (based on a small corpus that was collected solely for this demo; Figure 16.4); and a representation of the closeness to samples of the works of a random selection of different Dutch and Flemish authors (based on, on average, the first 11,000 tokens of one of their novels; Figure 16.5). The gender infobox is the result of assessing the proportion of male and female labels of 71 vectors that are in the vicinity of the vector that is based on the input, using cosine similarity. The other two infoboxes are based on the Dice coefficient (a metric that measures similarity) between the different (normalised) vectors that represent the style of the authors.

It should be noted that the models used are simplified, and that we do not make any scientific claims about the selection of genres and authors, or about the meaning of the output of the system. The only goal of this interface is to show what computational stylometry is, and provide a feeling for the type of information it uses and the type of output it produces.

16.3 The Readability Interface

Automatic readability prediction has a long and rich tradition. Research in the 20th century, fuelled especially by educational purposes, has resulted in a large number of readability formulas. Typically, these yield either an absolute score (Flesch, 1948; Brouwer, 1963) or a grade level at which a text is deemed appropriate (Dale and Chall, 1948; Gunning, 1952; Kincaid et al., 1975) and are based on shallow text characteristics such as average word and sentence length and word familiarity.

Over the years, many objections have been raised against these traditional formulas: their lack of absolute value (Bailin and Grafstein 2001), the fact that they are solely based on superficial text characteristics (DuBay 2004; DuBay, 2007; Davison and Kantor 1982; Feng et al. 2009; Kraf and Pander Maat 2009), the underlying assumption of a regression between readability and the modelled text characteristics (Heilman et al. 2008), etc. Furthermore, there seems to be a remarkably strong correspondence between the readability formulas themselves. When evaluating the performance of 12 readability formulas, of which 7 designed for English, 5 for Dutch and one for Swedish,

van Oosten et al. (2010) found strong correlations between the formulas, within a given language, but also across languages.

These objections have led to new quantitative approaches for readability prediction which adopt a machine learning perspective for the task. Advancements in these fields have introduced more intricate prediction methods such as Naïve Bayes classifiers (Collins-Thompson and Callan 2004), logistic regression (François 2009) and support vector machines (Schwarm and Ostendorf 2005; Feng et al. 2010; Tanaka-Ishii et al. 2010) – and especially more complex features. Rather than a sole reliance on superficial text characteristics, the added value of features measuring lexical complexity based on n-gram modelling (Schwarm and Ostendorf 2005; Pitler and Nenkova, 2008; Kate et al. 2010) or those relying on deep syntactic parsing (Schwarm and Ostendorf, 2005) have been corroborated repeatedly in the computational approaches to readability prediction that have surfaced in the last decade (Heilman et al. 2007; Petersen and Ostendorf, 2009; Nenkova et al. 2010). Features relating to semantics and discourse processing have proven more difficult to corroborate. While Pitler and Nenkova (2008) have clearly demonstrated the usefulness of discourse relations, the predictive power of these was not corroborated by Feng et al. (2010), for example. Especially for those features requiring deep linguistic processing, a lot still has to be explored (Collins-Thompson 2014).

In the readability interface, we present a re-implementation of several readability formulas and propose a new readability prediction system which does not only take into account these superficial text characteristics, but also relies on features grasping lexical complexity based on n-gram modelling and syntactic complexity based on deep syntactic dependency parsing.

16.3.1 General Text Characteristics

Once a text is provided, either by cut and paste or through file upload, we first present the user with some of the more general characteristics of the text. We include three length-related features that have proven successful in previous work (Nenkova et al. 2010; Feng et al. 2010; François and Miltsakaki 2012): the average word and sentence lengths and the percentage of polysyllabic words (i.e. words containing more than three syllables). We also incorporate two traditional lexical features: on the one hand, we provide the percentage of words also found in a Dutch word list with a cumulative frequency of 77% (or 'freq77').[2] On the other hand we also calculate the type token ratio (TTR) to measure the level of lexical complexity within a text.

All these characteristics are obtained after processing the text with a state-of-the-art Dutch preprocessor, Frog (Van den Bosch et al. 2007) and a designated classification-based syllabifier (van Oosten et al. 2010). Figure 16.6 illustrates how these general characteristics are presented to the user. It should be noticed that we also allow the user to actually highlight those words that contain more than three syllables or that are infrequent in Dutch.

16.3.2 Readability Judgement Based on Classical Formulas

Though many objections have been raised against the classical readability formulas, they remain popular and are still the go-to solution in many disciplines where a reader or author desires a first insight into text readability, e.g. corporate communication (Dempsey et al. 2012) or legislation (van Boom 2014). This is why, in a second step, we apply a number of readability formulas to the text which was entered by the user in the interface.

In essence, a readability formula is a mathematical formula intended for indicating the difficulty of a particular text. The formula typically consists of a number of variables, which are

[2] The list is based on a list ordered by descending frequency in a large newspaper corpus, i.e. the '27 Miljoen Woorden Krantencorpus 1995', which is available through the HLT agency at http://tst.inl.nl/en/producten

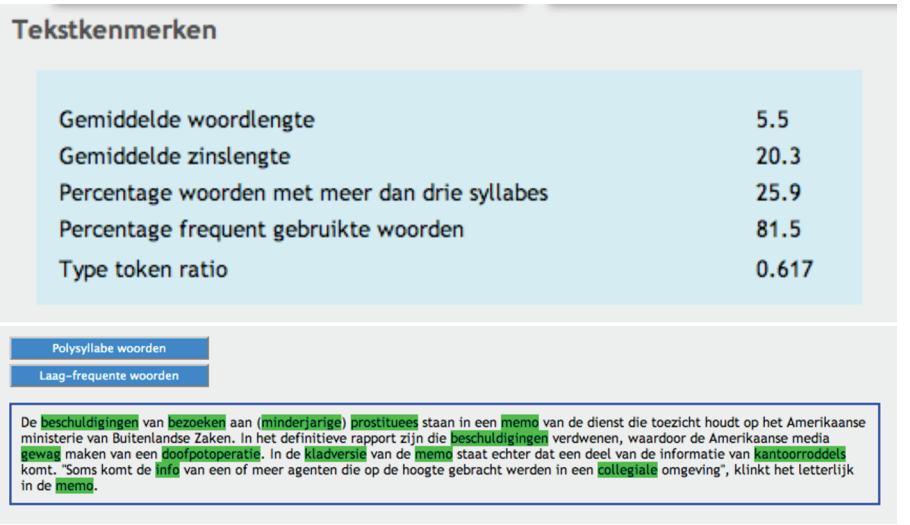

Figure 16.6: General text characteristics of the entered text.

characteristics of the text (as displayed in Figure 16.6) and constant weights. Besides the five general text characteristics that were introduced earlier (the average sentence length avgsentencelen; average word length avgwordlen; percentage of polysyllabic words, ppolysylword; freq77; and TTR), five additional variables are required for calculating all of the different formulas presented in the interface. These variables were derived using the same preprocessing toolkits as mentioned above and are listed below:

- avgnumsyl: average word length in number of syllables.
- psw: percentage of sentences per word.
- freq3000: percentage of words not on the Dale-Chall (1948) word list[3]
- avgpolysylsent: average number of words with three or more syllables per sentence.
- ratiolongword: ratio of words with more than six characters

These additional variables are not presented as such to the user. Instead, we display the results of the different formulas for a text which was entered by the user (see Figure 16.7). These formulas have been designed for Dutch (Douma, 1960; Brouwer, 1963; Staphorsius, 1994), English (Dale and Chall 1948; Flesch, 1948; Gunning, 1952; Senter and Smith, 1967; McLaughlin, 1969; Coleman, 1975; Kincaid et al., 1975) or Swedish (Björnsson, 1968). As van Oosten et al. (2010) have shown that there is a strong correspondence between the readability formulas intended for different languages, all readability formulas are displayed in the interface independently of the language they aim to model. The following readability formulas are displayed in the interface:
Dutch-language formula:

- Leesindex Brouwer ($195 - 2 \times$ avgsentencelen $- 67 \times$ avgnumsyl)
- Flesch-Douma ($207 - 0.93 \times$ avgsentencelen $- 77 \times$ avgnumsyl)
- CILT: Cito leesindex technisch lezen ($114 + 0.28 \times$ freq77 $- 12 \times$ avgwordlen)
- CLIB: Cito leesbaarheidsindex voor het basisonderwijs ($46 + 0.47 \times$ freq77 $- 6.6 \times$ avgwordlen $- 0.37 \times$ TTR $+ 1.4 \times$ psw)

[3] The Dale-Chall word list contains 3,000 of the most frequent words in the English language.

Leesbaarheidsformules ontwikkeld voor het Nederlands

NAAM	SCORE	SCHAAL
Leesindex A Brouwer	24.39	0 - ca. 120
Flesch-Douma	37.8	0 - ca. 120
CILT (Cito Leesindex Technisch Lezen)	69.72	50 - 100
CLIB (Cito Leesindex voor het Basis onderwijs)	54.75	0 - 100

Leesbaarheidsformules ontwikkeld voor andere talen

NAAM	SCORE	SCHAAL
Flesch Reading Ease	21.26	0 - ca. 120
Dale-Chall Reading Grade Score	18.68	0 - 16
Coleman-Liau Index	15.02	0 - 15
Flesch-Kincaid Grade Level	15.32	1 - 15
Gunning Fog Index	18.47	0 - 15
ARI (Automated Readability Index	14.51	1 - 15
SMOG (Simple Measure of Gobbledygook	16.22	1 - 15
LIX (Lasbarhetsindex Bjornsson)	49.88	0 - 100

Figure 16.7: Readability scores for entered text.

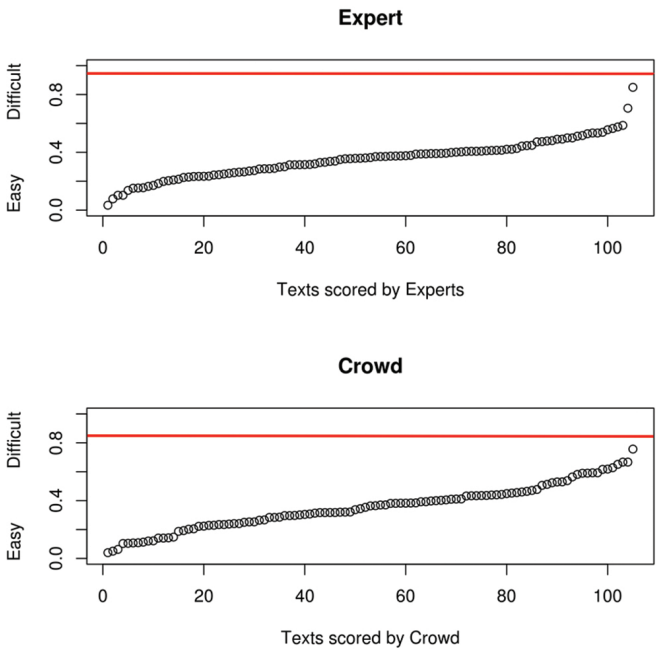

Figure 16.8: Hendi readability score of the text under consideration in comparison to the expert and crowd readability assessments of all texts in the training corpus.

English-language formula:

- Flesch Reading Ease (207 − avgsentencelen − 85 × avgnumsyl)
- Dale-Chall Reading Grade Score (0.16 × freq3000 + 0.05 × avgsentencelen + 3.6)
- Coleman-Liau Index (5.9 × avgwordlen − 0.3 × avgsentencelen − 16)
- Flesch-Kincaid Grade Level (0.39 × avgsentencelen + 12 × avgnumsyl − 16)
- Gunning Fog Index (0.4 × (avgsentencelen + ppolysylword))
- ARI: Automated Readability Index (4.7 × avgwordlen + 0.5 × avgsentencelen − 21)
- SMOG: Simple Measure of Gobbledygook: $\sqrt{}$(30 × avgpolysylsent) + 3.1

Swedish-language formula:

- Läsbarhetsindex Björnsson: avgsentencelen + ratiolongword

The last column in Figure 16.7 gives information on the scale on which the formulas are calculated. For some formulas (all English formulas except for Flesch Reading Ease; the Swedish formula; and the Dutch CLIB and CILT), a higher score applies to a more difficult text and a lower score to a more readable text; their slope is considered positive. For the other formulas, viz. Flesch Reading Ease, Flesch-Douma and Leesindex Brouwer, the situation is exactly opposite and the slope is considered negative.

16.3.3 Readability Prediction Based on Supervised Machine Learning

Given the many objections raised against the classical formulas, we also judge the readability of the entered text using the corpus-based readability prediction system developed by De Clercq et al. (2014). In order to compile the gold standard underlying this system, first a general-purpose corpus consisting of a large variety of text genres was compiled which was then assessed on readability. For the actual assessments, two web applications were designed to collect readability assessments for Dutch and English texts: one that is intended exclusively for language experts and one that is open to the general public. Both applications are available under the following link: http://www.lt3.ugent.be/en/tools/

Figure 16.8 gives an overview of these text scores assigned to texts by the experts and the crowd. The red line in both figures shows how our corpus-based readability prediction system scores the text compared to the other texts in the corpus.

Two flavours of the Hendi system have been integrated in this interface: a system which mainly relies on traditional and lexical text characteristics, and a second system which also integrates information representing the syntactic complexity of the text.

The former readability prediction system relies on a feature space of the traditional features mentioned above and lexical n-gram features which have proven to be good predictors of readability in previous work. Since we tried not to have presuppositions about the various levels of complexity in our corpus, a generic language model for Dutch was built based on a subset of the SoNaR corpus (Oostdijk et al. 2013). This subset contains only newspaper, magazine and Wikipedia material and should qualify as a generic representation of standard written Dutch. The language model was built up to an order of 5 ($n = 5$) with Kneser-Ney smoothing using the SRILM toolkit (Stolcke 2002). As features we calculated the perplexity of a given text when compared to this reference data and also normalised this score by including the document length, as seen in Kate et al. (2010). For more information on this system we refer the reader to De Clercq et al. (2014) and De Clercq and Hoste (2016).

In the latter system, syntactic information as displayed in Figure 16.9 is also taken into account. To this purpose we incorporated the parse tree features as first introduced by Schwarm and Ostendorf (2005) and that have proven successful in many other readability prediction studies

Name	Value
Average dependency tree depth	9.5
Average number of subordinating conjunctions	1.5
Average number of passive constructions	0.5
Average number of noun phrases	9.0
Average number of prepositional phrases	7.5
Average number of verb phrases	4.0

1.2. Other Formulas

Name	Value
Sentences with subordinating conjunctions	2
Sentences with passive constructions	2
Sentences with deep syntactic trees	0

Figure 16.9: Syntactic information calculated on the basis of the dependency tree of the sentence under consideration.

(Pitler and Nenkova 2008; Petersen and Ostendorf 2009; Nenkova et al. 2010; Feng et al. 2010). We calculate the parse tree height, the number of subordinating conjunctions and the ratios of the noun, verb and prepositional phrases. As an additional feature, we also include the average number of passive constructions in a text. The parser underlying these features is the Alpino parser (van Noord et al. 2013), a state-of-the-art dependency parser for Dutch.

As the parsing of the text may take some time, this calculation is performed offline and a pdf report is sent to the user as soon as the text is fully processed.

16.4 The Stylometry Interface

The Stylometry Machine Learning (ML) interface makes possible experiments following the full textcategorisation approach to stylometry: it allows the linguistic analysis of Dutch language documents, the extraction of features used regularly in the research literature, the creation of instances for ML experiments using these features, and the ML experiments themselves. We will describe here the different steps to use it in turn. The interface itself contains helpful hints, examples, and information as well. The system available through the interface has reduced functionality compared to the full stand-alone system, which is also available from the authors.

To use the interface on their data, users must first provide an email address in the appropriate field of the interface so that results can be sent to that address. Then the following procedure must be applied.

16.4.1 Step 1. Preparing and Uploading Data for Training

The goal of a supervised ML experiment is to use examples of some mapping to learn a model that generalises to independent similar data. For example, on the basis of a number of texts we know to have been written by Willem Elsschot and other texts written by other authors, we train a machine learning method to learn a model of the style of Elsschot. Afterwards we can test the accuracy of this model by applying it to texts that we did not use for training. The interface therefore makes a distinction between a Training run and a Testing run, and the user starts by uploading data for training.

Suppose we want to do a stylometry experiment predicting the gender of the author of tweets. We create a directory with two subdirectories (one for male, one for female), and put the 'train' tweets each in a separate file in their corresponding subdirectory. All files should be .txt files with

utf8 encoding. After creating a .zip file by compressing this directory (a directory that has as many subdirectories as classes – here, two – and with the texts belonging to each class in their corresponding subdirectory), this archive can be uploaded for training. After uploading, a result screen is presented indicating successful uploading and providing an identity number for further use.

16.4.2 Step 2. Defining the Experimental Parameters

To set up the way the uploaded data will be treated in building a model about style, several types of information have to be provided. First of all, a name has to be provided for the corpus (i.e. the data) that has been uploaded – e.g., 'Elsschot-1', or 'Gender-twitter', etc. This could for example be the name of the top directory in which you provided subdirectories with training texts.

Next, up to three 'analyses' can be provided. An 'analysis' in this context is a specific definition of the information that will be used to represent the text for the ML algorithm (the so-called document representation or instance definition). To define an analysis, the user selects a type, n-gram size, and frequency counting method. Analysis types supported are token (the tokenised words occurring in the text), character (the characters occurring in the text), lemma (the lemmatised tokens in the text), and pos (the part of speech, or grammatical category, of the words in the text). The n-gram size refers to the length of the sequences that we take into account; e.g. for characters, 'n' set to 3 would select all the character trigrams occurring in the text. A sentence such as 'Give me a break!' would result in the following character trigrams: '=Gi, Giv, iv=, =me, me=, =a=, =br, bre, rea, eak, ak=, =!='. Analogously, selecting n = 2 with tokens would result for the same sentence in the token bigrams '= Give, Give me, me a, a break, break !, ! ='. Additional information to be provided for each analysis is the frequency count type which can be absolute (how many times does a particular feature, for example the character trigram '=!=' occur in the document) or relative (what is the proportion of the occurrences of this feature in all the occurrences of all features in the document).

For each analysis specified, two datasets will be generated: one where document representations consist of binary vectors, and one where they consist of numeric vectors (where the numeric values are absolute or relative as selected by the user). In addition a dictionary is provided with the selected features for that experiment, their position in the document vector, and their frequency.

16.4.3 Step 3. Selecting the Features

The document representation defined in the previous step can be very large. In the 'filter' step, this set of features can be reduced to a manageable number on the basis of frequency, informativeness or a combination of both.

There are three filters that can optionally be selected. If none is selected, all features will be used. The total set filter allows the defining of a frequency band. For example, we might be interested in selecting the 10% most frequent features (set upper percentage to ten and leave lower percentage at 0), the 50% least frequent features (set upper percentage to 0 and set lower percentage to 50), or the middle band (in case one wants the features that are neither very frequent nor infrequent) – in this last case both thresholds could be set to 20, for example. (It is worth reminding that the term 'features' in this context refers to the items generated for the document representation, such as character trigrams or lemma bigrams.)

Not all features are equally relevant for distinguishing between classes. Statistical and information-theoretic methods such as chi-squared and information entropy can be used to analyse the degree to which a particular feature (e.g. the character trigram '=!=') can differentiate between the classes. The two remaining filters order the features according to relevance as defined by these methods and allow the selection of a percentage of these most relevant features.

All that remains to be done at this stage is indicating whether one wants document features (average word length, average sentence length, average number of syllables, number of hapax legomena, number of hapax dis legomena and readability) to be computed, which Machine Learning algorithm one wants to use, and which document representation (binary or numeric). By clicking start, the whole process will be activated and an ID generated.

The user will receive by email a zip file that contains all instance vectors for the analyses and filters chosen for the current training run. The email will also contain a unique identifier that is used as a link between the training run and any test run the user may want to perform in relation to this training run.

16.4.4 Step 4. Testing

With the identifier provided, the user can enter the Stylene machine learning interface again, this time with a test dataset submitted in the same format as the training data. The trained model will be applied on the test data provided and an analysis will be returned.

The user will receive by email a zip file that contains all the instance vectors that have been generated for this test run.

16.4.5 Using the Interface for Text Analysis Only (Optional)

In case the user is interested only in parsing their text(s), it is possible to go to the Frog parser interface, and submit an archive of texts (again following a zip archive format now with one directory of files to be analysed) that will then only be parsed. No ML models will be built in that case and with each input file in the archive a Frog output file will appear with the parsed input text. The Frog parser is also accessible from the Readability interface. The Frog parser used for this project is frozen at version 0.12.15 (c) ILK 1998 - 2012 to prevent compatibility issues in the future.

16.5 Conclusion

The Stylene project, funded by the Department of Economy, Science, and Innovation of the Flemish government, and executed by the department of Economy, Science and Innovation of the Flemish government (EWI), and executed by the CLiPS[4] and LT3[5] research groups, resulted in several resources, collected behind a single interface, that we hope will prove useful for different categories of users. People interested in the computational linguistics applications of stylometry and readability can analyse texts and be educated about the types of analysis that these research fields apply. Users in the digital humanities can test the automatic text categorisation approach to stylometry in a userfriendly interface suited for exploratory research. Whereas the first stylometry interface is based on simplified models, the readability interface and the machine learning of stylometry interfaces rely on stateoftheart software for Dutch. In addition, the interface provides an easy access to the stateoftheart Dutch text analysis software package Frog.

Acknowledgements

Apart from the authors, several people participated in the development of the Stylene system components and implementation. We gratefully acknowledge the contributions of Philip van

[4] http://www.clips.uantwerpen.be
[5] http://www.lt3.ugent.be

Oosten, Dries Tanghe, Peter Velaerts, Koen Vereeken, Guy De Pauw, and Vincent Van Asch. Herwig De Smet implemented most of the interface and the stand-alone stylometry system underlying the machine learning for the stylometry interface.

More information about the Stylene project can be obtained from
Prof. Dr. Walter Daelemans
CLiPS, Department of Linguistics, University of Antwerp
walter.daelemans@uantwerpen.be

References

Bailin, A. and Grafstein, A. (2001). The linguistic assumptions underlying readability formulae: a critique. Language & Communication 21(3):285–301.

Benjamin, Rebekah George. (2012). Reconstructing Readability: Recent Developments and Recommendations in the Analysis of Text Difficulty. Educational Psychology Review, 24(1):63–88.

Björnsson, C-H. (1968). Läsbarhet. Almqvist and Wiksell, Stockholm.

Brouwer, R. H. M. (1963). Onderzoek naar de leesmoeilijkheden van Nederlands proza. Pedagogische Studiën, 40:454–464.

Coleman, M. and Liau, T. L. (1975). A computer readability formula designed for machine scoring. Journal of Applied Psychology, 60:283–284.

Collins-Thompson, K. and Callan, J. (2004). A language modeling approach to predicting reading difficulty. Proceedings of the Human Language Technology Conference and the North American chapter of the Association for Computational Linguistics annual meeting (HLT - NAACL-2004), pp. 193–200.

Collins-Thompson, Kevyn and Jamie Callan. (2005). Predicting reading difficulty with statistical language models. Journal of the American Society for Information Science and Technology, 56:1448–1462.

Corpus Gesproken Nederlands (CGN) (2004). Nederlandse Taalunie. http://tst-centrale.org/images/stories/producten/documentatie/cgn_website/doc_Dutch/start.htm (Last accessed: June 2013).

Coulthard, Malcolm (2004). 'Author identification, idiolect, and linguistic uniqueness.' Applied linguistics 25.4: pp. 431–447.

Dale, E. and Chall, J. S. (1948). A formula for predicting readability. Educational research bulletin, 27:11–20.

Davison, A. and Kantor, R. (1982). On the failure of readability formulas to define readable texts: a case study from adaptations. Reading Research Quarterly 17(2):187–209.

De Clercq, O., Hoste, V., Desmet, B., van Oosten, P., De Cock, M., & Macken, L. (2014). Using the Crowd for Readability Prediction. Natural Language Engineering 20(3):293–335.

De Clercq, O. and Hoste, V. (2016). All mixed up? Finding the optimal feature set for general readability prediction and its application to English and Dutch. Computational Linguistics, 42:3.

Dempsey, S. J., Harrison, D. M., Luchtenberg, K. F., & Seiler, M. J. (2012). Financial Opacity and Firm Performance: The Readability of REIT Annual Reports. The Journal of Real Estate Finance and Economics, 45(2):450–470.

Douma, W. (1960). De leesbaarheid van landbouwbladen: een onderzoek naar en een toepassing van leesbaarheidsformules. Bulletin, 17.

DuBay, W. H. (2004). The Principles of Readability. Impact Information.

DuBay, W. H. (ed.) (2007). Unlocking Language: the Classic Readability Studies. BookSurge.

Feng, L., Elhadad, N. and Huenerfauth, M. (2009). Cognitively motivated features for readability assessment. Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL-2009), pp. 229–237.

Feng, L., Jansche, M., Huenerfauth, M. and Elhadad, N. (2010). A comparison of features for automatic readability assessment, Proceedings of the 23rd International Conference on Computational Linguistics Poster Volume (COLING-2010), pp. 276–284.

Flesch, R. (1948). A new readability yardstick. Journal of Applied Psychology, 32(3):221–233.

François, T. (2009). Combining a statistical language model with logistic regression to predict the lexical and syntactic difficulty of texts for FFL. Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop (EACL-2009), pp. 19–27.

François, T. and Miltsakaki, E. (2012). Do NLP and machine learning improve traditional readability formulas? Proceedings of the 1st Workshop on Predicting and Improving Text Readability for Target Reader Populations(PITR2012), pp. 49–57.

Gunning, R. (1952). The technique of clear writing. McGraw-Hill, New York.

Heilman, M. J., Collins-Thompson, K., Callan, J. and Eskenazi, M. (2007). Combining lexical and grammatical features to improve readability measures for first and second language texts. Proceedings of the Human Language Technology Conference and the North American chapter of the Association for Computational Linguistics annual meeting (HLT - NAACL 2007), pp. 460–467.

Heilman, M., Collins-Thompson, K. and Eskenazi, M. (2008). An analysis of statistical models and features for reading difficulty prediction. Proceedings of the 3rd ACL Workshop on Innovative Use of NLP for Building Educational Applications (EANL-2008), pp. 71–79.

Kate, R. J., Luo, X., Patwardhan, S., Franz, M., Florian, R., Mooney, R. J., Roukos, S. and Welty, C. (2010). Learning to predict readability using diverse linguistic features. Proceedings of the 23rd International Conference on Computational Linguistics (COLING-2010), pp. 546–554.

Kincaid, J. P., Jr., R. P. F., Rogers, R. L., and Chissom., B. S. (1975). Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel. Research branch report RBR-8-75, Naval Technical Training Command Millington Tenn Research Branch, Springfield, Virginia.

Klare, George. (1976). A second look at the validity of the readability formulas. Journal of reading behavior 8:159–152.

Kraf, R. and Pander Maat, H. (2009). Leesbaarheidsonderzoek: oude problemen, nieuwe kansen, Tijdschrift voor Taalbeheersing 31(2):97–123.

McLaughlin, G.H. (1969). SMOG grading - a new readability formula. Journal of Reading, pp. 639–646.

Nenkova, A., Chae, J., Louis, A. and Pitler, E. (2010). Structural features for predicting the linguistic quality of text: Applications to machine translation, automatic summarization and human-authored text. Empirical Methods in NLG, Lecture Notes in Articial Intelligence 5790, pp. 222–241.

Oostdijk, N., Reynaert, M., Hoste, V. and Schuurman, I. (2013). The construction of a 500-million-word reference corpus of contemporary written Dutch. Essential Speech and Language Technology for Dutch, Theory and Applications of Natural Language Processing, Springer, pp. 219–247.

Pennebaker, J. W. and Francis, M.E. (1996). Cognitive, emotional, and language processes in disclosure. Cognition and Emotion 10(6):601–626.

Pennebaker, J. W., Francis M.E., and Booth R.J. (2001). Linguistic Inquiry and Word Count (LIWC): LIWC2001. Mahwah: Lawrence Erlbaum Associates.

Pennebaker, J. W., Francis M.E., and Booth R.J. (2007). Linguistic Inquiry and Word Count (LIWC): LIWC2007. http://www.liwc.net.

Petersen, S. and Ostendorf, M. (2009). A machine learning approach to reading level assessment, Computer Speech & Language 23(1):89–106.

Pitler, E. and Nenkova, A. (2008). Revisiting readability: A unified framework for predicting text quality. Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP-2008), ACL, pp. 186–195.

Schwarm, S. E. and Ostendorf, M. (2005). Reading level assessment using support vector machines and statistical language models. Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL-2005), pp. 523–530.

Senter, R. J. and Smith, E. A. (1967). Automated readability index. Technical Report AMRLTR-66-220, University of Cincinnati, Cincinnati, Ohio.

Si, Luo and Jamie Callan. (2001). A Statistical Model for Scientific Readability. In Proceedings of the tenth international Conference on Information Knowledge Management, pages 574–576.

Staphorsius, G. (1994). Leesbaarheid en leesvaardigheid. De ontwikkeling van een domeingericht meetinstrument. Cito, Arnhem.

Stolcke, A. (2002). SRILM - an extensible language modeling toolkit, Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP-2002), pp. 901–904.

Tanaka-Ishii, K., Tezuka, S. and Terada, H. (2010). Sorting texts by readability, Computational Linguistics 36(2):203–227.

van Boom, W. (2014). Begrijpelijke hypotheekvoorwaarden en consumentengedrag, in T. B. en A.A. van Velten (ed.), Perspectieven voor vastgoedfinanciering (Congresbundel Stichting Fundatie Bachiene), Stichting Fundatie Bachiene, pp. 45–80.

Van den Bosch, A., Busser, G.J., Daelemans, W., and Canisius, S. (2007). An efficient memory-based morphosyntactic tagger and parser for Dutch, In F. van Eynde, P. Dirix, I. Schuurman, and V. Vandeghinste (Eds.), Selected Papers of the 17th Computational Linguistics in the Netherlands Meeting, Leuven, Belgium, pp. 99–114.

Van Halteren, H., Baayen, R.H., Tweedie, F., Haverkort, M. and Neijt, A. (2005). New Machine Learning Methods Demonstrate the Existence of a Human Stylome. In Proceedings of Journal of Quantitative Linguistics. pp. 65–77.

van Noord, G. J., Bouma, G., van Eynde, F., de Kok, D., van der Linde, J., Schuurman, I., Sang, E. T. K. and Vandeghinste, V. (2013). Large scale syntactic annotation of written Dutch: LASSY, Essential Speech and Language Technology for Dutch, Theory and Applications of Natural Language Processing, Springer, pp. 231–254.

van Oosten, P., Tanghe, D., and Hoste, V. (2010). Towards an Improved Methodology for Automated Readability Prediction. In Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., and Tapias, D., editors, Proceedings of the seventh International Conference on Language Resources and Evaluation (LREC'10), Valletta, Malta. European Language Resources Association.