PROCEEDINGS OF THE 11TH INTERNATIONAL CONFERENCE "LINGUISTIC RESOURCES AND TOOLS FOR PROCESSING THE ROMANIAN LANGUAGE"
26-27 NOVEMBER 2015

Editors:

Daniela Gîfu

Diana Trandabăț

Dan Cristea

Dan Tufiș

Organisers

Faculty of Computer Science
"Alexandru Ioan Cuza" University of Iași

Research Institute for Artificial Intelligence "Mihai Drăgănescu"
Romanian Academy, Bucharest

Institute for Computer Science
Romanian Academy, Iași

Under the auspices of the Academy of Technical Sciences

This volume was published with the support of
the Faculty of Computer Science,
"Alexandru Ioan Cuza" University of Iași

and

the National Authority for
Scientific Research and Innovation (ANCSI)

ISSN 1843-911X

PROGRAM COMMITTEE

Verginica Barbu Mititelu, Research Institute for Artificial Intelligence "Mihai Drăgănescu", Romanian Academy, Bucharest
Costin Bădică, Faculty of Automation, Computers and Electronics, University of Craiova
Tiberiu Boroș, Research Institute for Artificial Intelligence "Mihai Drăgănescu", Romanian Academy, Bucharest
Mihaela Colhon, Faculty of Mathematics and Natural Science, University of Craiova
Dan Cristea, Faculty of Computer Science, "Alexandru Ioan Cuza" University and Institute for Computer Science, Romanian Academy, Iași
Ștefan Daniel Dumitrescu, Institute of Artificial Intelligence "Mihai Drăgănescu", Romanian Academy, Bucharest
Corina Forăscu, Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iași and Research Institute for Artificial Intelligence "Mihai Drăgănescu", Romanian Academy, Bucharest
Daniela Gîfu, Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iași
Adrian Iftene, Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iași
Cătălina Mărănduc, Institute of Linguistics "Iorgu Iordan - Al. Rosetti", Romanian Academy, Bucharest and Faculty of Computer Science, "Alexandru Ioan Cuza" University
Rada Mihalcea, Computer Science and Engineering, University of North Texas, the United States of America
Alex Mihai Moruz, Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iași
Vivi Năstase, Fondazione Bruno Kessler, Trento
Ionuț Pistol, Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iași
Dan Ștefănescu, Faculty Audible (Amazon), Newark, NJ
Elena Isabelle Tamba, "A. Philippide" Institute for Romanian Philology, Romanian Academy, Iași
Cristiana Nicola Teodorescu, Faculty of Letters, University of Craiova
Diana Trandabăț, Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iași
Dan Tufiș, Research Institute for Artificial Intelligence "Mihai Drăgănescu", Romanian Academy, Bucharest
Marius Zbancioc, Institute for Computer Science, Romanian Academy, Iași

ORGANISING COMMITTEE

Anca Diana Bibiri, Department of Interdisciplinary Research - Human and Social Field, "Alexandru Ioan Cuza" University of Iași
Mihaela Colhon, Faculty of Mathematics and Natural Science, University of Craiova
Dan Cristea, Faculty of Computer Science, "Alexandru Ioan Cuza" University and Institute for Computer Science, Romanian Academy, Iași
Corina Forăscu, Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iași and Research Institute for Artificial Intelligence "Mihai Drăgănescu", Romanian Academy, Bucharest
Lucian Gâdioi, Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iași
Daniela Gîfu, Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iași
Adrian Iftene, Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iași
Diana Trandabăț, Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iași
Dan Tufiș, Research Institute for Artificial Intelligence "Mihai Drăgănescu", Romanian Academy, Bucharest

TABLE OF CONTENTS

FOREWORD

The edition of this year of the ConsILR conference is the 11th and, as usual, it concentrated on the advancements in enhance the Romanian language with new resources and improved processing tools. The contributions were clustered into 6 chapters (Speech Technology, Language Resources, Semantics, Text Analytics, Language Processing Tools, and Morphology and Syntax). The acronym used to name this series of conferences, comes from *Consorțiul de Informatizare pentru Limba Română* (Consortium for the Informatisation of the Romanian Language), thus remembering to all participants the initial goals of this initiative: from its inception, the conference was meant as a meeting place for linguists and computational linguists, but also for researchers belonging to the humanities, PhD students and master students in Computational Linguistics, all with a major interest in the study of the Romanian language from a computational perspective.

The series of ConsILR events started in the format of a workshop held every two years, but in 2010 we decided to turn it into an annual itinerary conference, in order to reach wider visibility, being addressed to researchers working on Romanian language from inside or outside Romania. This year, the ConsILR conference came back to Iași, from where it started 14 years ago. As always, it was organised under the auspices of the Romanian Academy and, additionally for this edition, under the auspices of the Academy of Technical Sciences of Romania (ASTR).

The traditional organisers of the Conference *Linguistic Resources and Tools for Processing the Romanian Language* are the Faculty of Computer Science of the "Alexandru Ioan Cuza" University of Iași and three institutes of the Romanian Academy: the Research Institute for Artificial Intelligence "Mihai Drăgănescu" in Bucharest, the Institute for Computer Science in Iași and the Institute of Romanian Philology "A. Philippide" in Iași. The venue of ConsILR-2015 was "Alexandru Ioan Cuza" University of Iași. Founded one year after the establishment of the Romanian state, by an 1860 decree of Prince Alexandru Ioan Cuza, under whom the former Academia Mihăileană was converted to a university, the University of Iași, as it was named at first, is the oldest university of Romania, and one of its advanced research and education institutions.

The invited conferences, not included into this volume, were delivered by Dan Tufiș, Horia-Nicolai Teodorescu, and Mihai Lupu, project assistant at the Vienna University of Technology. The lecture "ELRC – Translation for All", delivered by Acad. Tufiș, made a comprehensive presentation of the

latest initiative of the European Commission on coordinating language resources acquisition for all EU languages, in order to ensure high quality translation for public information services. The lecture ”Speech Technology for Master Students – Development of a Curricula”, presented by Prof. Horia-Nicolai Teodorescu, revealed the continuous updating of the MSc Speech Technology Curricula, based on the analysis of the students' performance and their feedback. In his invited talk, Dr. Mihai Lupu presented a view on language technology from the industrial perspective.

The multilinguality was considered in several papers: besides Romanian, languages such as Bulgarian, Spanish, English, French, Russian and old Romanian – written with Cyrillic characters, were addressed.

We hope that the quality of the selected papers makes the present volume, alongside the volumes from previous editions, an interesting source of information on what is happening in Romanian natural language scientific and industrial community, a collection of very useful articles for researchers and professors approaching the fields of AI, NLP and Computational Linguistics, as well as for students and anybody who is concerned with language use in the electronic media, either dedicated to Romanian language or being applicable to Romanian.

As in other editions, the complete program of the Conference and an electronic edition of the volume can be consulted online (at http://consilr.info.uaic.ro/2015/).

Iași, București - November 2015
The editors

CHAPTER 1
SPEECH TECHNOLOGY

ASPECTS OF THE SPEECH TRANSCRIPTION WITHIN THE SPOKEN LANGUAGE COMPONENT OF THE COROLA CORPUS

VASILE APOPEI[1], LUMINIȚA HOARȚĂ CĂRĂUȘU[2], DOINA JITCĂ[1]

[1] *Institute of Computer Science, Romanian Academy, Iaşi branch*
[2]*Faculty of Letters, "Alexandru Ioan Cuza" University of Iaşi*

vasile.apopei@iit.academiaromana-is.ro, lumicarausu@yahoo.com, doina.jitca@iit.academiaromana-is.ro

Abstract

This paper aims to present some aspects of the discursive-pragmatic transcription for Romanian spoken language and of the coherence of sentence structure resulting from these transcriptions. The approach is based both on existing papers about Romanian spoken language corpora and on problems encountered in the transcription of spoken language component for the COROLA corpus.

Keywords: coherent sentence structure, disfluency, discursive-pragmatic transcription.

1. Introduction

In the last twenty years, several "Romanian corpuses of spoken language" (Cărăușu, 2013; Ionescu-Ruxăndoiu, 2002; Dascălu Jinga, 2002) have been developed, which in fact are impressionist transcripts of recordings of the dialogues / spontaneous utterances, made by using various transcriptions symbols. Based on these transcription conventions, functional discursive-pragmatic transcripts are done for spoken language corpora, transcripts marking on the text with symbols the disfluencies (discontinuities) and prosodic elements that accompany oral communication. By disfluencies, in this paper, we mean those events and phenomena that occur in spoken language (spelling, hesitations, thinking pauses, reformulations, interruptions, overlapping talk, unintelligible sequences of words, etc.), which produce discontinuities and unnatural prosodic phrasing.

Even if these corpora are made for the express purpose of being used in computer processing (Ghido, 2004), transcription conventions allow more analysis for Pragmatic and Communication Theory and very little for the Syntax of spoken language. Moreover, so far to our knowledge, for Romanian these "corpora" are not aligned with utterances which have been transcribed and do not have an XML structure to facilitate searches for intonation and discursive events and their correlation with syntactic and lexical structures.

Worldwide, there are initiatives to facilitate access to information contained in the discursive transcripts of the spoken language corpora made with the conventions proposed by Du Bois *et al.* (1993), Jefferson (2004). In this way, in the last years, in the projects CHILDES (Child Language Data Exchange System) and CLARIN (Common Language Resources and Technology Infrastructure), new transcription standards using XML technology and software tools have been developed that take the linguistic resources made for different purposes and made by annotators with different experience in order to integrate them into wider corpora of spoken and written language.

2. An analysis of discursive-pragmatic transcripts from syntactic perspective

Aurelia Merlan (1998), in an analysis of the syntax of spoken Romanian language, reveals some issues which arise in the communication act and which cause discontinuities in achievements of sentence with less rigorous syntactic structure, affecting the syntactic coherence of sentences. Among these aspects we found again those ones that create the transcription problems for the spoken language component of the COROLA corpus.

- Sentences are the result of cooperative behavior of the participants to dialogue (e.g. dialogues teacher / student):

> A: deş' pi ieste prepoziţiyee, + şî din-cauzâ câ iesti + cuvînt.
> B: nenoţional. ((Merlan, 1998) citing Adrian Turculeţ)

- Participants to the dialogue dispute their transmitter role for developing a coherent sentence:

> A: Era bini sî-fii
> B: liber
> A: sî-aibî vacanţă chiar.
> B: Ca în Germania
> A: măcar o săptămînă.
> B: di-Anu Nou, înainti di Anu Nou două săptămîni cu
> A: Da, da.
> B: plată li dău la toţi, +cîti două săptămîni, (Merlan, 1998)

- One of participants to the dialogue intervenes to express curiosity, approval to those exposed:

A: am chiulit. +++şi-am fugit ↑ + ă ↑ ++ în sala unde dumneaei dansa fără niciun afiş. + nu era anunţat ↑
B: unde?
A: PLIN. + PLIN de oameni ↑ ++ într-o sală ↑ + la teatrul evreiesc cred că era ↑ ++ era printr-o ↑ + într-o sală ↓ +++ PLINĂ. (Hoarţă Cărăuşu, 2013)

- One of participants to the dialogue interferes for the continuation of statements (the intervention takes place on the thinking pause)

A: pentru un băiat ↑ +++ mai ales ↑ mai ales pe vremea aceea ↑ + în cel mai bun caz erai ↑ +
B: prim solist [al ↑
A: aveai] rolul principal ↑ + şi erai ↑ ++ ă ↑ + erai ↑ + prinţul din lacul lebedelor (Hoarţă Cărăuşu, 2013)

Besides syntax aspects of spoken Romanian language presented in (Merlan, 1998), at the Institute of Computer Science we began an analysis on a set of transcripts from the Romanian spoken language corpus presented in (Hoarţă Cărăuşu, 2013). The analysis aims at the possibility to recover the syntactic coherence of the sentences by using the symbols for prosodic indications. Preliminary analysis revealed that these indications correspond to boundaries of prosodic phrases, but some indications of "final descending melodic contour" break syntactic phrases in text segments that have no sentence syntactic structure.

A: da. ++ face lucruri. +++ pe care nimeni nu poa' să le facă.

The point after the word 'lucruri', used to mark the final downward intonation, followed by a longer silent pause, accurately marks the intonation phrase but creates problems in restoring syntactical coherence to the associated text.

The next example reveals the same problem, the interruption of the sentence generated by the intervention of speaker B.

A: este REALMENTE un SPIRIT. ++ nu numai materie ↑ + ci spirit ↑ ++ şi-n acelaşi timp ↑ + este o CIZELARE ↑ ++ o cizelare fără ↑ + fără ↑ + ă ↑ + fără sfârşit ↑ + a acestei păsări care devine la un moment dat ZBOR.
B: ZBOR
A: şi nu pasăre.

The points after the words 'SPIRIT' and 'ZBOR', used to mark the final downward intonation contours, followed by a longer silent pause, accurately marks the intonation phrases but create problems in restoring syntactical coherence to the associated text.

Another drawback of the discursive-pragmatic transcriptions is generated by sequences not translated for reasons of legal compliance to the personal data processing. Lack of transcription for these word sequences creates difficulties in automatic speech-to-text alignment.

3. Importing discursive-pragmatic transcriptions

Even if the "Romanian spoken language corpora", are designed having in mind analyzes of the Pragmatic and Communication Theory, we believe that these transcripts can be used in automatic speech-to-text alignment in order to develop corpora for training systems in automatic speech recognition. With few changes to the transcription conventions, these corpora can be imported and used for training prosodic phrasing modules for text-to-speech systems.

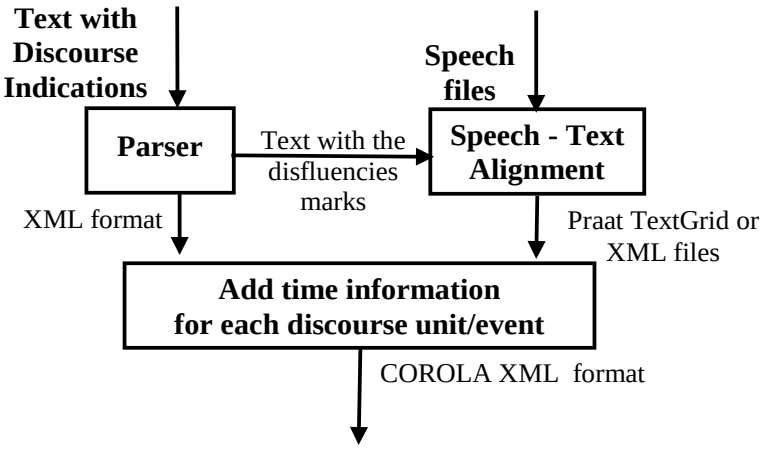


Figure 1. The diagram for importing discursive-pragmatic transcriptions into the COROLA corpus

The analysis made on the set of transcripts extracted from the Romanian spoken language corpus presented in (Hoarță Cărăușu, 2013) reveals that some of the symbols used to mark events and phenomena that occur during speech are of interest also in the context of the spoken language component of the COROLA corpus. Further development of the software tools that allow importing the existing information in these corpora towards COROLA and exporting from the COROLA format towards the format used by linguists create premises for interdisciplinary

researches of interest to specialists in computational linguistics and communication theory.

The parsing process of the discursive-pragmatic transcripts consists, on the one hand, in restoration of the syntactic phrases along with disfluencies that occur during their utterances and, on the other hand, in the extraction of indications for prosodic elements (intonation, rhythm, silent pauses). Syntactical phrases are aligned with the corresponding voice recording and processing results are stored in a TextGrid or XML file type. In a later stage, the information about prosodic elements will be added to the XML file aligned with voice recordings from the COROLA corpus.

4. Conclusions

The discursive-pragmatic transcription of corpora, when they are transcripts of impressionist type, depends heavily on the experience and subjectivity of the annotators. Therefore, we believe that an alignment of these transcripts with corresponding audio recordings (text-to-speech alignment) would allow the verification of these transcripts by experienced annotators and the development of software tools to identify records that do not have complete / correct transcripts.

The aspects of spoken Romanian language syntax, related to syntactic coherence of statements, presented in section 2, leads us to propose the usage, in the COROLA corpus annotation, of a special tag for syntactical sentence.

Acknowledgements

The research presented in this paper has been developed within the Institute of Computer Science of the Romanian Academy, Iasi branch. The selection of the sentence and discursive annotation of the corresponding utterances was made by Luminiţa Hoarţă Cărăuşu.

References

Apopei, V. (2014). About prosodic phrasing of the Noun Phrases in speech. In *Proceedings of the Romanian Academy*, Vol. 15, Number 2/2014, pp. 200-207.

Apopei, V., Păduraru, O. (2015). Towards Prosodic Phrasing of Spontaneous and Reading Speech for Romanian Corpora. In *Proc. of the 8th Conf. Speech Technology and Human - Computer Dialogue* (SpeD), Bucharest, Romania, Oct. 14-16, 2015, IEEE (to appear on IEEEXplorer).

Dascălu Jinga, L. (2002). Corpus de română vorbită (CORV). Eşantioane, Bucureşti, Oscar Print.

Du Bois J. W., Schuetze-Coburn S., Cumming, S., and Paolino D. (1993). Outline of discourse transcription. In Talking data: Transcription and coding in *Discourse Research*, eds. Jane A. Edwards and Martin D. Lampert, pp. 45-89. Hillsdale, NJ: Erlbaum.

Ghido, D. (2004). Aspecte ale transcrierii limbii române vorbite în vederea prelucrării computerizate, în *Aspecte ale dinamicii limbii române actuale*, Actele Colocviului Catedrei de limba română a Facultăţii de Litere din Universitatea Bucureşti (27-28 noiembrie 2002).

Hoarţă Cărăuşu, L. (2013). Corpus de limbă română vorbită actuală nedialectală, Editura Universităţii "Alexandru Ioan Cuza", Iaşi.

Ionescu-Ruxăndoiu, L. (2002). Interacţiunea verbală în limba română actuală. Corpus (selectiv). Schiţă de tipologie, Bucureşti, Editura Universităţii din Bucureşti.

Jefferson, G. (2004). Glossary of transcript symbols with an introduction. In *Conversation, Analysis: Studies from the First Generation*, G.H. Lerner (ed.), Amsterdam: John Benjamins, pp. 13-31.

Jitca D., Apopei V., Jitca M. (2009). *The F0 Contour Modelling as Functional Accentual Unit Sequences,* International Journal of Speech Technology, Volume 12, Issue 2-3, pp. 75-82.

Jitcă, D., Apopei, V., Păduraru, O. and Maruşca, S. (2015). Transcription of Romanian intonation, in S. Frota & P. Prieto (eds), *Intonational in Romance*, Oxford University Press, pp. 284-316.

Merlan, A. (1998). Sintaxa şi semantica - pragmatica limbii române vorbite: discontinuitatea, Editura Universităţii "Alexandru Ioan Cuza", Iaşi.

P. Boersma, D. Weenink, Praat, www.fon.hum.uva.nl/praat/

Selkirk, E. (2005). Comments on intonational phrasing in English, Prosodies: With Special Reference to Iberian Languages, S. Frota, M. Vigario, and M.J. Freitas (eds.), Berlin: Mouton de Gruyter, pp. 11-58.

TEI (2015). TEI P5: Guidelines for Electronic Text Encoding and Interchange, edited by C.M. Sperberg-McQueen and Lou Burnard for the ACH-ALLC-ACL under the supervision of the Technical Council of the TEI Consortium.

Zienkowski, J., Östman, J.-O. and Verschueren, J. (eds.) (2011). Discursive Pragmatics, University of Antwerp / University of Helsinki.

INSTRUMENTS FOR PROCESSING ACOUSTIC DATA IN THE PROJECT THE CONTRASTIVE ANALYSIS OF ROMANIAN AND SPANISH INTONATION. A SOCIOLINGUISTIC APPROACH (*SOROES*)

ANCA-DIANA BIBIRI[1], MIHAELA MOCANU[1], LIVIU-ANDREI SCUTELNICU[2, 3], ADRIAN TURCULEȚ[1]

[1] *Department of Interdisciplinary Research – Humanities and Social Sciences, "Alexandru Ioan Cuza" University of Iaşi*
[2] *Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iaşi*
[3] *Institute of Computer Science, Romanian Academy, Iaşi Branch*

anca.bibiri@gmail.com, mihaelamocanuiasi@yahoo.com, andrei.scutelnicu@info.uaic.ro, aturcu@uaic.ro

Abstract

This paper aims to present the tools employed to achieve a contrastive analysis of Romanian and Spanish language intonation, at the methodological level, in the project *The Contrastive Analysis of Romanian and Spanish Intonation. A Sociolinguistic Approach* (SoRoEs). The project addresses the sociolinguistic dimension of intonation in Romanian and Spanish languages, aimed at identifying specific elements and lines of continuity that we recorded in both Romance languages. One goal of the research is to achieve a digitized corpus for both languages taken into consideration. It will provide support and tools for further contrastive analysis, getting a real insight into the structure and dynamics of intonation from an interdisciplinary perspective. Our analytical approach is to establish the sociolinguistic patterns specific to the two Romance varieties types of utterances investigated and their distribution in space.
To identify the central tendency in our data set we have to determine what type of inferential statistical methods we should use. We will use tests of significance to make reasonable extrapolations from our corpus using SPSS (Statistical Package for Social Sciences).

Keywords: contrastive analysis, prosodic patterns, Romanian language, sociolinguistics/sociophonetic methodology, Spanish language.

1. Introduction

Sociolinguistic profile of a nation implies establishing a complex hierarchy of languages (and language variants) also from the perspective of social functions that govern behavior in groups and societies. The accurate knowledge of this reality is indispensable in linguistic decision making (Cazacu, 1973).

Recently, among the studies dedicated to prosody there are few accomplishments concerning sociolinguistic approaches. Many investigations are restricted to dialectal differences, while age, level of education, sex, speaking style and individual habits are ignored. One project that has provided evidence for sociolinguistic variation in intonation is the project English Intonation in the British Isles (Grabe, Nolan, Post, 1997-2002, http://www.phon.ox.ac.uk/files/apps/IViE/), a research that takes into account as variables the dialect, speaking style, gender and individual speaker habits.

As for this sociolinguistic approach, our study is also important for several reasons. Firstly, it is worth reminding ourselves that although the sociolinguistic works are nowadays understood as a great dimension of linguistic research, the study of intonation is, in fact, a far cry from it although there are some studies which provide insight into its field. In seeking to understand the large scale of problems of a sociolinguistic approach to intonation, some scholars made significant pioneer contributions to it: Moreno-Fernández (1998), Martín-Butragueño (2011). Moreover, it seems that cross-gender variation has received more attention: López-Bobo and Cuevas-Alonso (2014), Vermillion (2001), Daly & Warren (2000), and for Romanian language, some tentative studies: Tiugan (1977), Panaite and Turculeț (2011).

Secondly, as pioneer researches, some of these studies have not always faced seriously the approach and the methods by which it was studied. Cepeda's approaches (Cepeda and Roldán 1995; Cepeda, 1998) allow some misunderstandings, as the corpus was too extensive. Moreover, as we could see, some scholars fitted the interpretation of the results into the descriptive statistical frameworks or based their work on written rather than spoken language. As Baker (2010) points out '[…] the fact that corpus studies […] have used written rather than spoken texts means that such studies are unable to reveal very much about the origin of an innovation'.

On the other hand, this is coupled with clear problems which stemmed from the great variability (phonetic and phonological) that the intonation could offer due to pragmatic variables. Since the sociolinguistic variation has not been a major concern of the researchers of prosody, it shows that there is necessary a new sociolinguistic methodology, for a study which has to signal opportunities for minimal responses, such as the discovering of those variables that reflect the intonation variation, tracing the paths that lead to such divergence and widening our understanding of it.

This article aims to a broad presentation of the project SoRoEs and the instruments for data analysis recorded in sociolinguistic dialectal surveys, current state of the research, results and prospects for the two fields under discussion.

2. Methodological approach

The SoRoEs is a project that will record a series of oral materials in audio format (and, part of them also in video format) intended to provide a basis for further in-depth studies of the two Romance languages: Romanian and Spanish. Researching diastratic variation (vs. diatopic, diaphasic, diamesic) is the subject of sociolinguistics, especially of sociophonology/socioprosody. Diatopic features can function as diastratic and diamesic markers (Turculeț, 1999); in socioprosody intonational patterns should be correlated (reflected in speech melody/F0 curve) with socio-cultural status of the speaker. In his study of American English, William Labov (1972), based on surveys in New York and Martha's Vineyard, concludes that the source of the evolution of the language is hypercorrectness, promoted by medium strata (between low and high). For example, a hypercorrection utterance would retrieve the final contour (CT) extended on the final unstressed vowel, induced possibly in the school: students are instructed to raise their voices at the end of the question; also, there are situations of 'contamined'contours by changing discoursive strategy by subjects who prove a certain tendency.

Sociolinguistic survey involves simultaneous use of a number of informants (five subjects for each social variable are enough) with different ages and professions, being born in respective places, reffering as Haugen appreciate, 'the intensive study of a population' (Haugen, 1971). Since this project is based on sociolinguistic aspects of intonation, in defining the subjects for prosodic survey, we consider three basic social variables: level of education, age and sex. Thus, we consider male and female subjects, with primary, secondary and university education, for age category there are 3 groups (20-25 years, 35-40 years, over 50 years), taking into account that: 'Girls are often said to use a more 'expressive' intonation than boys, who 'play it cool'; 'expressive' intonation refers to the two factors of more rises and wider key. The fundamental frequency of women's voices is, of course, physiologically conditioned to be higher than men's' (Cruttenden, 2000).

The methodology and database are common to both languages. For Spanish, a part of the corpus has already been recorded in the project El Atlas Interactivo de la Entonación del Español (four points of survey – Oviedo, Pola de Siero, Santander, Vigo) of the six that we intend to study (besides the 4 mentioned, Cabezón de la Sal and Pontevedra); for Romanian, will be recorded fixed questionnaire (containing different types of declarative sentences, interrogative, imperative, vocatives, interjections in semantic and pragmatic contexts), adapted to the diatopic features of Romanian language (10 points of survey in cultural centers of the country: Bucureşti, Iaşi, Cluj, Braşov, Timişoara, Craiova, Constanţa, Botoşani, Sibiu, Oradea).

In addition, a spontaneous corpus (including various statements as, for example: yes-no questions, WH-questions, echo questions, imperatives, vocatives, etc.) lasting 40 to 60 minutes will be recorded.

Our research includes videos, a multimodal analysis method that examines the relationship between face expression and the posture of the subjects and the general context. Integrating images 'in motion' is a methodological challenge, so it is necessary to ensure clarity during recording and positioning the camera so to capture the best of the assembly.

3. Acoustic processing of the recorded data

Acoustic analysis is a 'maximalist' one, including the main physical features of each vowel: the duration, maximum intensity (sound energy) and fundamental tone (F0). Based on the melodic profiles resulted as the evolution of F0 in time and the sound energy, obtained from the analysis of each sentence, prosodic contour is performed according to the laryngeal frequency average of the speaker.

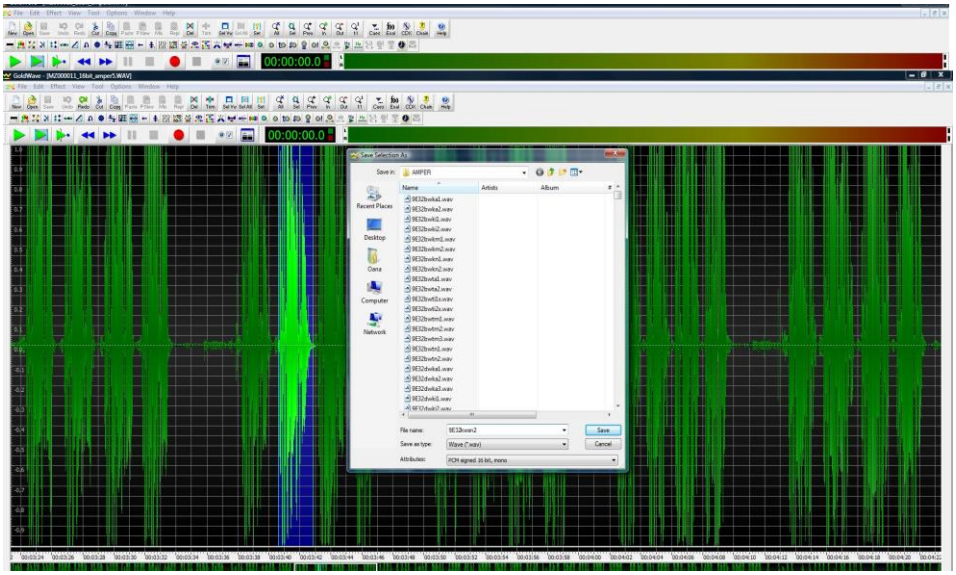


Figure 1. Delimitation and labeling the statements in GoldWave

Based on intonational contours (which bear phonostylistic and attitudinal characteristics and prosodic markers of the geographical origin of the speaker) generated in various graphics, comparative research is done. This approach aims at

the documentation of specific prosodic patterns of Romance varieties for the types of recorded sentences and their diatopic distribution.

The corpus is recorded with a frequency of 44,000 Hz, with a resolution of 16 bits in wav file format. Statements are recorded in digital format (wav file extension – Waveform Audio File) and there are analyzed acoustic using software tools. The review process successively through several stages:

Changing the sampling frequency of the sound wave from 48 kHz to 16 kHz (GoldWave[1]);

- Delimitation statements and labeling according to the questionnaire used with GoldWave (as shown in Figure 1);
- The segmentation and labeling of the vowel elements: assisted by the software PRAAT[2] (Paul & Boersma, 2013) – there follows the segmentation and labeling vowel elements (in the case of diphthongs the two vowel marks go together), based on oscillograms, spectrograms and by hearing (Figure 2):

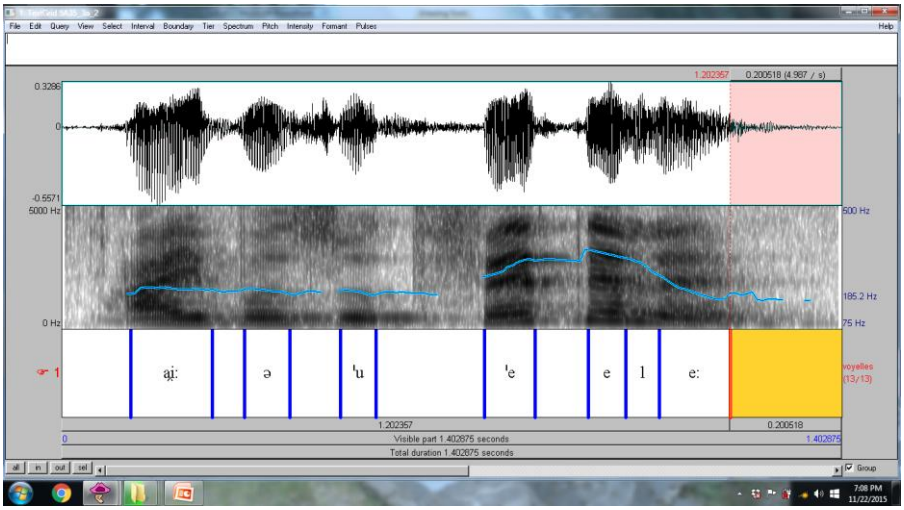


Figure 2. The segmentation and labeling vowel elements in PRAAT

[1] https://www.goldwave.com/
[2] http://www.praat.org/

- The conversion of TextGrid file in TXT in which are found physical correlates of vowels: duration, intensity and fundamental frequency (F0 – for the three points of the vowel) as shown in Figure 3:

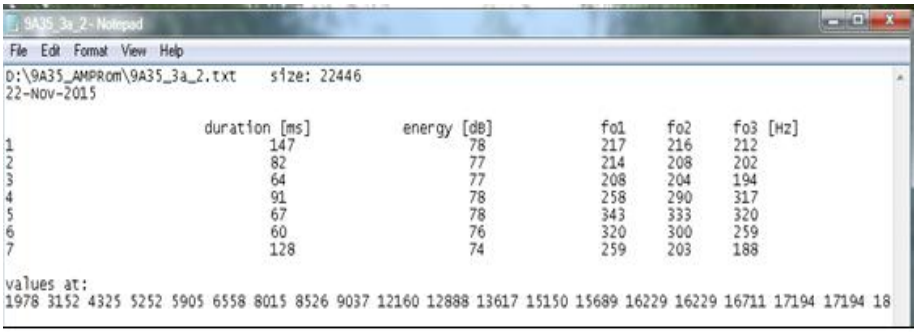


Figure 3. Duration, intensity and F0 of each vowels of the statement in *.txt* file

To identify the central tendency in our data set we have to determine what type of inferential statistical methods we should use. We will use tests of significance to make reasonable extrapolations from our corpus using SPSS[3] (Statistical Package for Social Sciences).

4. Conclusions and future work

Through this project we aim to realyze a database (http://soroes.ro/) that will include corpora recorded and acoustic processed files for the two Romance languages; this will work as a resource for research carried out by specialists in linguistics, sociolinguistic, pragmatic, psycholinguistic and cultural fields. Digitized corpora stimulate various contrastive analysis of Romance intonation, allowing a real insight into the structure and dynamics of intonation from an interdisciplinary perspective.

Acknowledgements

This work was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS – UEFISCDI, project number PN-II-RU-TE-2014-4-2298.

[3] https://www.apponfly.com/en/ibm-spss-statistics-standard

References

Baker, P. (2010). Sociolinguistics and Corpus Linguistics. Edinburgh: Edinburgh University Press.

Boersma, P., Weeninck, D. (2013). PRAAT: doing phonetics by computer. Version 5.3.51, retrieved 2 June 2013 from http://www.praat.org

Cazacu, B. (1973). Un aspect al cercetării interdisciplinare: Sociolingvistica, in *Fonetică și dialectologie*, vol. VIII, București.

Cepeda, G., Roldán, E. (1995). La entonación del habla femenina de Valdivia, Chile: su función comunicativa, gramatical y expresiva, in *Estudios Filológicos* 30, 107-123.

Cepeda, G. (1998). El movimiento anticadencial en la entonación del español de Valdivia, in *Estudios filológicos*, 33, 23-40.

Cruttenden, A. (2000). Intonation, Cambridge: Cambridge University Press.

Firchow, E. S. (eds.) (1971). Studies by Einar Haugen: Presented on the occasion of his 65th birthday. De Gruyter Mouton.

Labov, W. (1972). Sociolinguistic Patterns. U. of Pennsylvania Press, Spanish translation *Modelos Linguisticos*. Madrid: Editions de Catedra. French translation, *Sociolinguistique*. Paris: Editions de Minuit.

López Bobo, M. J., Cuevas Alonso M. (2014). Estratificación sociolingüística de la entonación cántabra: la variable sexo, in *Fonética Experimental, Espacio Europeo de Educación Superior e Investigación*.

Moreno Fernández. F. (1998). Estudio sociolingüístico de la entonación, en: *Oralia* 1, pp. 95-117.

Martín Butragueño, P. (2011). Estratificación sociolingüística de la entonación circunfleja mexica, in P. Martín Butragueño (ed.), *Realismo en el análisis de corpus orales: primer coloquio de cambio y variación lingüística*. México.

Panaite, O., Turculeț, A. (2012). Diastratic Particularities of Speech Intonation used in Focsani, in *Analele Universităţii din Craiova*, Seria Ştiinţe filologice, Limbi străine aplicate, an VII, nr.1-2, pp. 211-225.

Tiguan, M. (1977). Sociolinguistic analysis of a phonetical variable, în RRL, XXII, nr.4, pp. 431-444.

Turculeț, A. (1999). Introducere în fonetica generală și românească, Casa Editorială *Demiurg*, Iași, p. 150.

Vermillion, P. (2001). The perception and production of intonational meaning by British men and Women. M. Phil dissertation, Queen Mary and Westfield College, University of London.

Warren, P., Daly., N. (2000). Sex as a factor in rises in New Zealand English, in J. Holmes (ed.) *Gendered Speech in Social Context: Perspectives from Gown and Town*, Wellington: Victoria, pp. 99-115.

CHAPTER 2
LANGUAGE RESOURCES

HYDRA FOR WEB: A MULTILINGUAL WORDNET VIEWER

BORISLAV RIZOV[1], TSVETANA DIMITROVA[1], VERGINICA BARBU MITITELU[2]

[1] *Institute for Bulgarian Language, Bulgarian Academy of Sciences*
[2] *Research Institute for Artificial Intelligence "Mihai Drăgănescu", Romanian Academy*

boby@dcl.bas.bg, cvetana@dcl.bas.bg, vergi@racai.ro

Abstract

This paper presents Hydra for Web – a web interface for wordnets (and lexical-semantic databases with similar relational structure). Hydra for web is built on top of Hydra – an open source tool for wordnet development – and is a single page application with a simple GUI. It has two modes – single and parallel – for visualisation of the language correspondences of searched words. The main focus of the paper is on the user interface, its functionalities and some issues regarding the ongoing process of adapting the resources that are currently visualised – the Princeton WordNet, the Romanian wordnet, and the Bulgarian wordnet.

Keywords: lexical databases, wordnet, web interface, open source tool

1. Introduction

Wordnet is a lexical-semantic database firstly created for English – the Princeton WordNet (cf. Miller, 1995; Fellbaum *et al.*, 1998). Synonymy is the main relation between the words in wordnet, and synonyms (in this paper termed 'literals') are organised in unordered sets called synonym sets (synsets) that are considered to reflect (psycho)linguistic concepts (Miller, 1990). For example, each of the two members of the synset {break dance:1, break dancing:1} is considered a literal. Except for one or more literals, a synset obligatorily contains a definition (for example, for the synset {break dance:1, break dancing:1}, the definition is '*a form of solo dancing that involves rapid acrobatic moves in which different parts of the body touch the ground; normally performed to the rhythm of rap music*'). A synset may also contain examples (one or more phrases or short sentences) to illustrate the usage of the concept. Synsets are interlinked via different conceptual relations for hypernymy/hyponymy (e.g., the hypernym of the synset {break dance:1, break dancing:1} is the synset {dancing:1, dance:1, terpsichore:1, sallation:1}), antonymy, meronymy, holonymy, and others. Synsets can also be related via morphosemantic relations (agent, event, result, etc., e.g., the concept expressed by the synset {break

dance:1, break dancing:1} is event of the concept expressed by the synset {break dance:2, break-dance:1, break:45}), derivational relations (derived/derivative), etc. (for details, see Miller, 1995; Fellbaum *et al.*, 1998).

The Princeton WordNet (PWN) includes only open class words, i.e., nouns, verbs, adjectives, adverbs. Nouns and verbs obligatorily have hypernyms (except for the roots of the trees organizing these parts of speech), while adjectives often have derivational relations and the relation <similar_to> linking them to their near synonyms, e.g., {stunning:1} is <similar_to> {beautiful:1}. The relation <derived>/<derivative> connects adverbs to the adjective from which they are derived, e.g., the synset {beautifully:1, attractively:1} is linked via the <derived> relation to the two synsets {beautiful:1} and {attractive:1} – these relations in fact hold between the literals of a synset.

Almost all wordnets (Romanian included) developed after the PWN model contain only words of these four parts of speech. The Bulgarian wordnet (BulNet), however, contains also closed class words (pronouns, prepositions, conjunctions, particles, interjections; cf. Koeva, 2010). Moreover, adverbs have an obligatory relation <category_member> linking them to synsets that define specific concepts of time, place, manner, frequency, etc.

The information in wordnets is very complex and is encoded in a (strictly) relational format. Due to this complexity, wordnets need tools for flexible, yet simple visualisation. In addition, the tools used for the creation of wordnets and the visualisation of the complex information also have to consider the relational character of the data. Hydra for web is such a tool. What is more, it can also show parallel wordnet data where wordnets with the same identification numbers for their elements can be visualised in parallel (for this purpose, the wordnets use the ILI – inter-lingual lexical index, as defined by Vossen, 2002).

2. Hydra and Hydra for web

Hydra for web[5] is built on top of Hydra – an open source tool for wordnet development, whose features were discussed in other papers (Rizov, 2008; 2014), so we will highlight here in brief only those that are relevant to Hydra for web. Hydra provides an API for access to any semantic network of the wordnet type (lexical-semantic relational databases). Hydra is implemented in Python, using the platform independent GUI library Tkinter, and has been extensively used for the development of the BulNet. The data is managed by a MySQL server.

Hydra is quite convenient for parallel working on (one or more) wordnet(s) because it allows users to access – for editing and query purposes – any number of wordnets simultaneously and to build complex queries to extract data in the available wordnet by using the information about wordnet hierarchy and relations (for example, one can search for all noun synsets in the database (presumably it may contain any

[5] The tool can be used at: http://dcl.bas.bg/bulnet/.

number of wordnets for any language) that are connected via <undergoer> relation to all verb synsets or only to verb synsets that contain specific literals or that are classified as cognitive or communicative, etc. verbs via a set of semantic prime labels; etc.). Parallel wordnets can be synchronised to allow simultaneous visualisation of the equivalent synsets in different languages.

Hydra is also useful for team working because it allows concurrent access by multiple users. The changes in the database are made available to all users right after they are made (plus given information about the user that has made the change and the time when the change has been made), and this is a feature that is kept in Hydra for web – if appropriate access is given, all the changes in a wordnet database can be visualised on the web right after they are made (as it works now for the BulNet). The Hydra for web tool is a web interface GUI implementation that uses Hydra as backend. The web interface, in fact, depends for most of its functionalities on Hydra.

In the next section, we will present the user interface and its functionalities together with a discussion on some compatibility issues between the databases, hence with the PWN database.

*3. **User interface and functionalities***

Hydra for web supports two modes – a single wordnet mode and parallel wordnets mode. Currently, it gives access to the Princeton WordNet (PWN) 3.0 (Fellbaum *et al.*, 1998), the Bulgarian wordnet (BulNet) 3.0 (Koeva *et al.*, 2004), and the Romanian wordnet (RoWN) (Tufiş *et al.*, 2013).

Hydra for web's interface can be localised and is currently available in English, Bulgarian, and Romanian, i.e., in the languages of the wordnets which are visualised.

Figure 1. Search for 'clasic' in the RoWN (Romanian interface)

The names of the relations and other elements were translated into Bulgarian and Romanian (the part of speech (pos), and language markers (en, bg, ro) are still kept in English, as well as most of the relations in the RoWN, from which we chose to translate only those occurring in current Romanian linguistic literature, e.g., *hiponim*, *hiperonim*, *implicație*).

The window has a top panel for switching the wordnets to be viewed. It currently allows for the options of a single wordnet, and three pairs of wordnets in the parallel mode, namely BulNet vs. PWN, BulNet vs. RoWN, and RoWN vs. PWN.

With a single query, users can search into databases of different language wordnets. The search panel is present in both single and parallel wordnets modes and it allows queries for an exact match of a word string – a single word such as 'clasic' in Romanian as shown on Fig. 1, or a multiword unit such as 'classical music' in English that is found in PWN and RoWN on Fig. 2.

The non-exact match search returns any synset where the word is found. For example, the search for 'pas' (see Fig. 3) returns different synsets from the English, Bulgarian and Romanian wordnets databases, including multiword units such as *pas de deux, pas de trois*, etc.

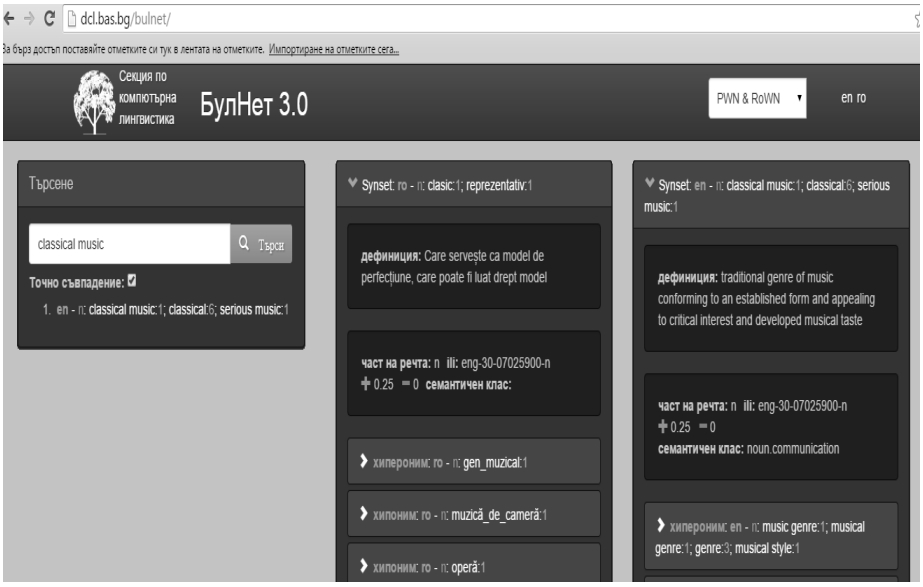


Figure 2. Search for 'classical music' and findings in RoWN and PWN (Bulgarian interface)

To limit the results shown, the search respects word (string) boundaries, i.e., the user can search only for whole words but not part of the words (as this often would return more than a hundred results; however, this option is available in Hydra for the

purposes of wordnet development and some of the more complex query options might be implemented in Hydra for web in the future).

In case of multiword units written with a hyphen, searching for one of the words in the unit allows the user to find the multiword expressions as well: see the synset containing *Nord-Pas-de-Calais* found when searching for *pas* in the Fig. 3 (see also the multiword units returned when searching for the word 'ballet' in Fig. 4).

Figure 3. Non-exact match search for 'pas' (English interface)

There were issues with presentation of the language data with respect to word boundaries with multiword units without hyphens. In RoWN, the multiword units were represented as one string (with an underscore between components, as in {balet_clasic:1}; this was done for reasons of compatibility with corpus processing strategy (especially tokenisation) and tools available for this), e.g., {balet_clasic:1} vs. {classical ballet:1).

However, unless the users know this convention, they will not find multiword units with the non-exact match search (without writing the exact form with the underscore); e.g., when searching for all the words (in synsets) with ‘dans’, the user will find only synsets featuring the exact match (‘dans’), but not {dans_modern:1, dans_contemporan:1}, {dans_din_buric:1}, etc. (this is not true for PWN and, eventually, BulNet, as the search for ‘classical’ will return also {classical ballet:1}).

Thus, as the tool does not allow for searching for parts of a word string, the user is supposed to know the exact form of what they are searching for (and this is

inappropriate for users who are not familiar with the conventions adopted during the RoWN development, but want to check some terms in the RoWN database). Consequently, we decided to discard the underscores with multiword units in the RoWN database that are used with the Hydra for web. Moreover, the consistency of the three databases with each other is another reason for our decision.

3.1. Single wordnet mode

The single wordnet mode consists of two panels – the search panel to the left of the screen (where the user can search for a word) and the synset view panel of the selected word, to the right of the screen. The search returns the synsets that contain searched for literals in all languages in the database. The right panel displays exactly the synset selected. For example, the search for 'pas' returns all synsets with 'pas', including synsets in Bulgarian and Romanian, and expressions like *pas de deux*, etc.

3.2. Parallel wordnets mode

The parallel wordnets mode has three panels, with the second and the third panel hosting the parallel wordnet views – see Fig. 4. Each of the two wordnet panels shows the correspondences of the synset selected (in the selected languages). In this way, the user can search for a word in English, e.g., 'ballet' and access the parallel synsets in the BulNet 'балет', and in the RoWN – 'balet', as shown in Fig. 4.

Figure 4. Search for 'ballet' in English and displaying the Bulgarian and Romanian correspondents of the selected English synset (Romanian interface)

Hydra for web is also available on a small width (mobile), where the panels are ordered successively – the search panel first and the synset views second and third (see Fig. 5).

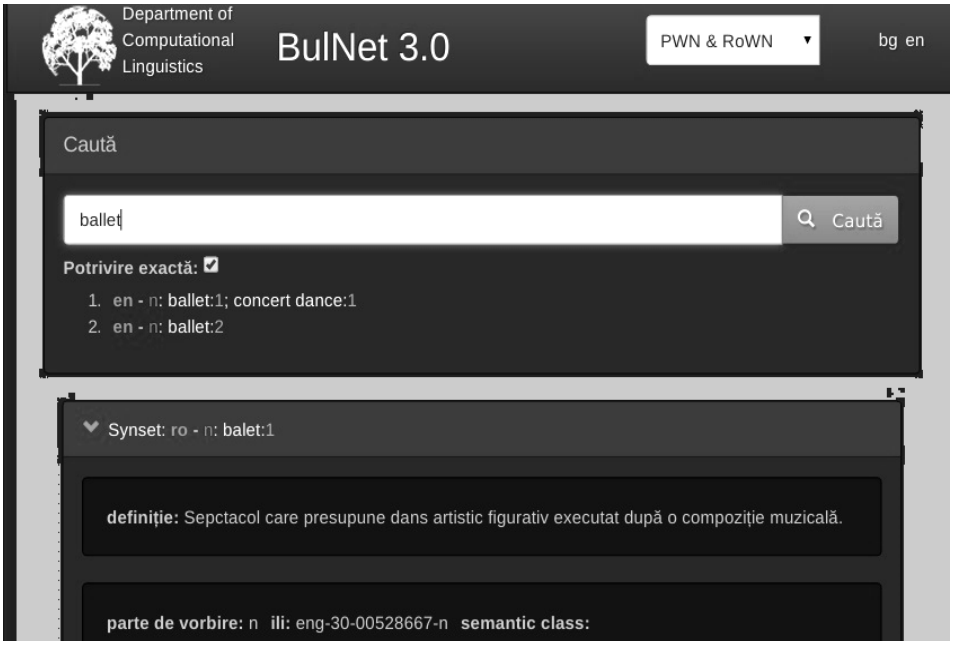


Figure 5. Hydra for web – small width (Romanian interface)

3.3. Visualisation of the wordnet data

The elements of the synset structure are visualised in a predefined order. The literals (synonyms) are first, the definition comes second. The relations are given in a predefined order and are distinguishable by colour as seen on the web (at http://dcl.bas.bg/bulnet/): hypernyms and hyponyms are in blue, the other relations are in white, pos is in orange beside the set of literals. Information about pos, ILI, sentiment scores according to SentiWordNet (Elusi and Sebastiani, 2006), and semantic class is stacked together.

The information in the relations is processed according to the synset's ILI. Thus, the current synset and the synset with the same ILI are marked with the arrow bullet turning red and pointing down – in Fig. 6, the two synsets in Bulgarian {готвач:1} and Romanian {bucătar:1} that are visualised after a query for the noun 'cook', are linked via <derivative> and <agent> relations to the synset {готвя:2, сготвя:2, сготвя:2, приготвям:2, приготвя:2} in Bulgarian and its Romanian equivalent {găti:3} in RoWN.

The visualization is recursive as every relation linked to another synset (<hypernym>, <holo_part>, etc.) is expandable in the same way as the root one (thus, eventually, the user can get a whole tree). Information like POS, ILI, etc. is available immediately, and the relations are loaded by means of AJAX query, but without blocking the UI.

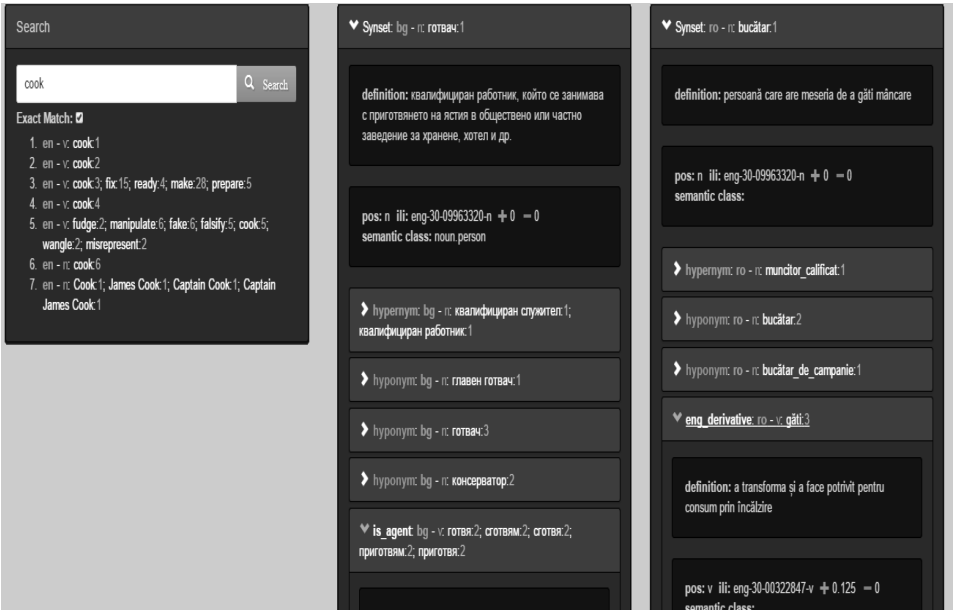


Figure 6. Synsets with the same ILI

When uploading different wordnets, issues of compatibility between relations also arise. There are certain relations that are missing in PWN, but are available in the BulNet, such as the obligatory <category_member> with the adverb synsets, as well as <bg_derivative> with Bulgarian synsets that are not present in English.

The BulNet also has information on the literal level such as the derivational patterns with nouns and verbs that are linked through morphosemantic relations (cf. Dimitrova *et al.*, 2014), some literal notes (for the verb aspect with the literals in the verb synset, or literal notes, with the relation <lnote> at the level of the literal where various information about form, usage, style, etc. is given). In addition, the BulNet contains all parts of speech, while PWN and RoWN cover only open class words, therefore the synset for the closed class words is available only in BulNet (without having a parallel synset).

Some relations in the RoWN imported from PWN are considered rather English-specific and they were prefixed with "near": e.g., the relation <usage_domain> from PWN was imported as <near_usage_domain> in RoWN as it may not hold for other languages (including Romanian), e.g., the literal {veg:1} in the synset {vegetable:1, veggie:1, veg:1} has a relation <usage_domain> with the synset {United

Kingdom:1, UK:1, Britain:1}, informing users of the language of the fact that this word is used in the territory named by the words in the latter synset; however, in the RoWN the relation was transferred at the synsets level: it holds between the synset corresponding to the one to which {veg:1} belongs, namely {legumă:1, zarzavat:1}, and the one corresponding to the {United Kingdom:1, UK:1, Britain:1}, namely {Marea Britanie:1}; however, it is clearly not the case that the words *legume* and *zarzavat* are used in *Marea Britanie*. Relations involving literals are language-specific and, thus, they do not (necessarily) hold in other languages (in this case Romanian), where they are automatically transferred.

To keep track of such cases, they are transferred into the RoWN with a prefix: "near-". In BulNet, some of the inappropriate relations were manually deleted or corrected but others were left as they are in PWN (with the option to be corrected later).

However, there were other relations' names in the RoWN that differed from the PWN and the BulNet which, in fact, encoded the same relation (<hypo> for hyponymy, etc.) and they were changed to their original name.

4. Other tools

Currently, there are more than 75 wordnets[6] (covering more than 40 languages including wordnets for minority languages) which are used in a growing number of applications.

All these wordnets use different tools for development and a number of web interfaces for browsing wordnet databases (such as Wordvis, Mexidex, etc.). There are also many web tools (including many online dictionaries) which use wordnet (especially PWN) as a database (e.g., Bee Dictionary; LookWAYUp; a2zDefined, cozyenglish, among others), though they do not provide access to all the information about the relations, as Hydra for web does.

There are also popular user interfaces that visualise wordnet relationships as graphs (such as Wordvis (Vercruysse and Kuiper, 2013)) which, however, do not support a parallel view of two or more wordnet language databases and do not always visualise the whole information. The PWN web tool WordNet Search 3.1[7] is useful for viewing the information of the PWN lexical-semantic trees, but it does not give access to all the available information (for example, morphosemantic relations are not part of it) and does not support a parallel view for different languages.

5. Implementation

Hydra for web is built with Node.js[8] and the web application framework for Node.js Express[9]. It is a single page application and uses one of the most popular HTML,

[6] http://globalwordnet.org/wordnets-in-the-world/
[7] http://wordnetweb.princeton.edu/perl/webwn
[8] Node.js® is a JavaScript runtime: https://nodejs.org/

CSS and JS frameworks – Bootstrap[10]. The application is themed in Slate from Bootswatch.[11] Bootstrap made easy the GUI to be responsive, and so it is mobile friendly. For the html rendering, the very clean and elegant JADE[12] template engine is used.

Many of the tasks in the GUI are solved in the client with the use of Knockout.js[13] framework. It uses declarative bindings, dependency tracking and provides automatic UI refresh. The wordnet data retrieval is made by means of the Wordnet Service. The retrieval uses AJAX and is completely asynchronous (non-blocking).

Wordnet service is a RESTful webservice written in Python and Twisted. The service uses the Hydra API to extract the information from the wordnet database. The service's API provides requests for searching and extracting the objects from the database. It is also useful for retrieving the neighbours of a particular wordnet object by all the relations (hypernyms, hyponyms, etc.) and its correspondent synsets in the other languages.

6. Applications

Hydra for web can be used for queries into different wordnets and for viewing parallel wordnets (or any lexical resource following similar relational structure) from every place, computer, phone or other device with internet connection. Parallel data can be used for comparative lexical and other linguistic studies.

One obvious application is as a multilingual dictionary. The list of results (single words and multiword units) returned also contains information about other (synonym) words and the part of speech of the resulting words.

7. Conclusions and future work

The paper presented the beta version of a web visualisation tool, Hydra for web, while outlining its user interface and functionalities for the general user of the information from three (for now) lexical-semantic databases: English, Bulgarian and Romanian.

There is still a lot of information that is part of different wordnets that can be made viewable depending on the wish of the respective developers (for example, derivational patterns, additional notes on the level of synsets and literals, etc.).

An option for composing more complex queries can be implemented by exploiting the wordnet hierarchy and relations to see the related data in parallel wordnets, though the philosophy for the tool at this stage is to keep it simple.

---

[9] http://expressjs.com/
[10] http://getbootstrap.com/
[11] https://bootswatch.com/
[12] http://jade-lang.com/
[13] http://knockoutjs.com/

Links to other resources (lexical, semantic, etc.) can also be implemented – the SentiWordNet (Esuli and Sebastiani, 2006) was already deployed (in the figures in this paper, the plus/minus signs for positivity and negativity sentiment scores can be seen in the synsets, beside the information about pos, ILI, and semantic class).

Different options for saving the results for later editing and analysis can be also made available depending on the developers' preferences and the resources' licenses.

Acknowledgements

Part of the work reported in this chapter is carried out within the joint project "Enhanced Knowledge Bases for Bulgarian and Romanian" of the Institute for Bulgarian Language, Bulgarian Academy of Sciences, and the Research Institute for Artificial Intelligence "Mihai Drăgănescu" of the Romanian Academy.

References

Dimitrova, T., Tarpomanova, E., Rizov, B. (2014). Coping with Derivation in the Bulgarian Wordnet. In: *Proceedings of the Seventh Global Wordnet Conference*, Tartu, Estonia, pp. 109-117.

Esuli, A., Sebastiani, F. (2006). SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining. In: *Proceedings of Language Resources and Evaluation (LREC)*, pp. 417-422.

Fellbaum, C. D. (ed.) (1998). *Wordnet: An Electronic Lexical Database.* Cambridge, MA: MIT Press.

Koeva, S. (2010). Bulgarian Wordnet – Current State, Applications and Prospects. In: *Bulgarian-American Dialogues*, Sofia: Academic Publishing House, pp. 120-132.

Koeva, S., Tinchev, T, Mihov, S. (2004). Bulgarian Wordnet – Structure and Validation. *Romanian Journal of Information Science and Technology*, 7(1-2), pp. 61-78.

Miller, G. A. (1990). WordNet: An on-line lexical database. *International Journal of Lexicography* 3, 4 (Winter 1990), pp. 235-312.

Miller, G. A. (1995). Wordnet: A Lexical Database for English. *Communications of the ACM*, November 1995, 38(11), pp. 39-41.

Rizov, B. (2008). Hydra: A Modal Logic Tool for Wordnet Development, Validation and Exploration. In: *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, pp. 1523-1528.

Rizov, B. (2014). Hydra: A Software System for Wordnet. In: *Proceedings of the Seventh Global Wordnet Conference*, Tartu, Estonia, pp. 142-147.

Tufiş, D., Barbu Mititelu, V., Stefanescu, D., Ion, R. (2013). The Romanian Wordnet in a Nutshell. *Language Resources and Evaluation*, December 2013, 47(4), pp. 1305-1314.

Vercruysse, S., Kuiper, M. (2013). WordVis: JavaScript and Animation to Visualize the WordNet Relational Dictionary. *Advances in Intelligent Systems and Computing*, 179, pp. 137-145.

Vossen, P. (2002). WordNet, EuroWordNet and Global WordNet. *Revue française de linguistique appliquée*, 1/2002 (Vol. VII), pp. 27-38.

AN APPROACH FOR INTERCONNECTING LEXICAL RESOURCES

LIVIU-ANDREI SCUTELNICU[1,2], ANCA-DIANA BIBIRI[3], DAN CRISTEA[1,2]

[1]*Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iaşi*
[2]*Institute of Computer Science, Romanian Academy, Iaşi Branch*
[3]*Department of Interdisciplinary Research – Humanities and Social Sciences, "Alexandru Ioan Cuza" University of Iaşi*

liviu.scutelnicu@info.uaic.ro, anca.bibiri@gmail.com, dcristea@info.uaic.ro

Abstract

Nowadays, great efforts are being made to enhance the utility of linguistic resources in applications related to language processing by interconnecting them. In this study we focus on Romanian, one of the European languages that has still to make big steps until being considered well resourced, from the point of view of interconnection of linguistic resources. As a case study, we refer to the tip-of-the-tongue (TOT) problem, i.e. how to help a human to remind a word that is not immediately reachable in memory, a problem that has received some attention recently. The Romanian resources used in this study are: RoWordNet, DEX-Online, the COROLA corpus and a collection of semantically organised words. Our solution considers first a unification of the representation format of the initial lexical resources, followed by their combination in view of making possible a navigation process that would not be concerned with some specific representation of data. The application of our method to the TOT problem makes a step forward towards building a tool that would help people getting fluency and continuity of ideas in communication. Seen more generally, the interconnection method proposed can be useful to researchers in natural language processing and linguists interested in combining different lexical resources.

Keywords: lexical resources, WordNet, corpus, dictionary, interconnection of lexical resources, natural language processing.

1. Introduction

The existence of lexical resources is of utmost importance for language technology. Acquiring and developing the basic tools for creation, structuring, exploitation and maintenance of language resources depends on the datasets specific to the respective language that varies from simple lists of words to complex resources, such as dictionaries, annotated corpora, grammars, treebanks, thesauri, ontologies, language models, etc. In the same time, the involvement of more types of resources in complex applications makes compulsory their easy interconnection. The

heterogeneity of language resources obliges to integration and interoperability (Chiarcos *et al.*, 2012) in order to be reusable for many applications and available to the NLP community.

In our paper we propose a methodology of making compatible different linguistic knowledge bases and exemplify this approach with a number of existing resources for Romanian language. The immediate application is to find a solution to the tip-of-the-tongue (TOT) problem, but the approach could be useful in other application settings that impose the combination of heterogeneous resources.

2. Lexical resources

Our lexical resources are monolingual in this study, namely representing the Romanian language. Lexical resources are databases structured in many ways: word lists, dictionary entries, synsets, etc. More precisely, in our study the following 4 resources have been used, each with a different structure:

- RoWordNet[13] – the Romanian WordNet (Tufiş and Cristea, 2002) (Tufiş *et al.*, 2013) is created following the model of Princeton WordNet (Fellbaum, 1998). This database contains nouns, verbs, adjectives and adverbs grouped in sets of cognitive synonyms called synsets, each expressing a distinct linguistic concept. Every synset has a unique identifier and groups a set of lexicals paired with their sense IDs; on this basis, the sense number of the searched/target word in the respective synsets can be extracted. The synsets for nouns and verbs are placed in a conceptual hierarchy by hypernym/hyponym relations;
- COROLA – the Computational Representative Corpus of Contemporary Romanian Language is a resource under construction now as a priority project of the Romanian Academy. At the end of the project (2017), this corpus, built in collaboration by the Research Institute in Artificial Intelligence in Bucharest and the Institute of Computer Science in Iasi is supposed to include texts totalizing 500 million Romanian words acquired from a broad range of domains (politics, humanities, theology, science, literature, etc.) and covering all literary genres (prose, poetry, theatre, scientific texts, journalistic texts, etc.) (Bibiri *et al.*, 2015). In order to acquire its running form, the texts (obtained on the base of written protocols from our providers), after discharging the formatting ballast, passed through a processing chain which included minimum: sentence segmentation, tokenization, lemmatization, and part of speech tagging. In our research we have used about 130 million words (out of the 200 million which make the corpus at present);

[13] http://www.racai.ro/en/tools/text/rowordnet/

- DEX-Online[14] – an online source-free dictionary created by voluntary contribution and merging a collection of sense definitions for 75,000 entries extracted from prestigious dictionaries of the Romanian language;
- A semantic lexicon (Gîfu, 2010) – a vocabulary containing 6000 words grouped by grammatical categories (nouns, verbs, adjectives and adverbs) and semantic classes (politics, religion, family, etc.).

Each of these resources are available (under specific licences) in the XML format.

3. Obtaining homogeneity in the representation of lexical resources

"Building up an infrastructure of language resources only makes sense if these resources are standardized." (Teubert, 1997). The development of reusable resources for interchanging and open-ended retrieval tasks is always dependent on the homogeneity of their representation (Sinclair, 1994). Thus, standardised methods and specifications can be applied for building links/bridges between the resources taken into consideration.

As mentioned, our resources are coded in XML, but this common format is too general to ensure compatibility. The idea is to rewrite the resources onto a form that could be accessed by the same interrogation code. Let's note, for the time being, that such goals have been formulated long time ago.

Text Encoding Initiative (TEI), for instance, has proposed an inventory of features most often deployed for computer-based text processing and has formulated recommendations about suitable ways of representing these features. The idea was to facilitate processing of different types of resources by computer programs, and the loss-free interchange of data amongst individuals and research groups using hardware platforms or application software. However, TEI is only a near-standard, not a standard in itself. More towards a standard is the Lexical Markup Framework (LMF). This is the ISO/TC37 standard for natural language processing (NLP) and machine-readable dictionary lexicons[15].

Both LMF and TEI model lexical material at a deep representational detail. However we want more than that since our primary intention is the interconnection of different lexical material in an easy way. Here are the main features we want to acquire:

- if two resources are to be connected, simply "merge" their contents, where "merge" mean putting the information together but not repeating common information and remaining specific there were differences occur. The differences should be reflected by specific feature names;

[14] https://dexonline.ro/

[15]http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=37327

- we should be able to represent variations in word forms, alternate orthography, diachronic morphology, etc. if the primary resources include such data;
- expression of restrictions in querying the merged resource should be made possible by applying various filtering criteria;
- if the lexicographic data is hierarchical (for instance, a sense of a dictionary entry contains, apart from a definition and examples, also sub-senses), then this should be reflected in the representation and in processing by recursive searches. Here is an example: "give me the definition neighbouring sphere of depth 2 of the word *captain*" should be solved by: "take all senses of the entry *captain* and form the list of words in the corresponding definitions, then for each of them take all their senses and collect again words in their definitions".

By applying these criteria, we implemented the "merge" operation by analysing the contents of each XML-coded linguistic resource presented in input to determine both their common and uncommon parts. For example, in the resources mentioned in the previous section some fields of the XML representation, such as definition, lemma and part of speech do exist in all resources and, as such, they were coded only once. The other tags, which code specific content, remain in each resource.

The operation of "merge" was implemented by retrieving the information from each XML resource as follows:

- first, we identified the common parts from the XML resources and inserted them into a new XML tag. In Figure 1, this is the *entry* tag, which contains the definition, the part of speech and other tags that are identical in all resources involved in the merge process;
- for tags that are not identical in all resources, new tags were created. Each of them contains, under a name that identifies the resource, specific data from the initial resource. In Figure 1, for exemple, for RoWordNet the specific tags are: ID, SENSE, DOMAIN, SUMO, ILR, inserted under the new tag *wn*;
- for the COROLA corpus, we extracted contexts that contain the target word (entry) and its neighboring words (4 proceeding and 4 following). This ressembles concordance, helping to know the use of the entry word in specific contexts.

```
<?xml version="1.0" encoding="UTF-8"?>
<entry POS="n" content="abate" id="1" def="Titlu dat superiorului unei abații">
  <dex>
   <definition>@ABÁTE^1,@ $abați,$ #s. m.#  @2.@ Titlu
    onorific acordat unor preoți catolici; persoană care poartă acest titlu. - Din #it.#
    @ab(b)ate.@</definition>
  </dex>
  <wn>
   <ID>ENG30-09754404-n</ID>
   <SENSE>1.1</SENSE>
   <DOMAIN>religion</DOMAIN>
   <SUMO>Position+</SUMO>
   <ILR>ENG30-10675876-n<TYPE>hypernym</TYPE></ILR>
  </wn>
  <corpus>
   <entry id="1">
    <W Case="oblique" LEMMA="lucrare" MSD="Ncfsoy" POS="NOUN" id="19.5">lucrării</W>
    <W Case="direct" LEMMA="patrologie" MSD="Ncfsry" POS="NOUN" id="19.6">Patrologia</W>
    <W Case="oblique" LEMMA="Graeca" MSD="Npfpon" POS="NOUN" id="19.7">Graeca</W>
    <W Case="" LEMMA="," MSD="COMMA" POS="" id="19.8">,</W>
    <W Case="direct" LEMMA="abate" MSD="Ncmsry" POS="NOUN" id="19.9">abatele</W>
    <W Case="oblique" LEMMA="Migne" MSD="Npfpon" POS="NOUN" id="19.10">Migne</W>
    <W Case="" LEMMA="avea" MSD="Vaip3s" POS="VERB" id="19.11">a</W>
    <W Case="" LEMMA="colecta" MSD="Vmp" POS="VERB" id="19.12">colectat</W>
    <W Case="" LEMMA="240" MSD="M" POS="NUMERAL" id="19.13">240</W>
   </entry>
   <entry id="2">
    <W Case="" LEMMA="pagină" MSD="Y" POS="ABBREVIATION" id="27.14">P.</W>
    <W Case="" LEMMA="genitiv" MSD="Y" POS="ABBREVIATION" id="27.15">G.</W>
    <W Case="" LEMMA=")" MSD="RPAR" POS="" id="27.16">)</W>
    <W Case="direct" LEMMA="al" MSD="Tsfsr" POS="ARTICLE" id="27.17">a</W>
    <W Case="oblique" LEMMA="abate" MSD="Ncmsoy" POS="NOUN" id="27.18">abatelui</W>
    <W Case="" LEMMA="junior" MSD="Y" POS="ABBREVIATION" id="27.19">J.</W>
    <W Case="" LEMMA="pagină" MSD="Y" POS="ABBREVIATION" id="27.20">P.</W>
    <W Case="oblique" LEMMA="Migne" MSD="Npfpon" POS="NOUN" id="27.21">Migne</W>
    <W Case="" LEMMA="," MSD="COMMA" POS="" id="27.22">,</W>
   </entry>
```

Figure 1. Example of a XML entry after merging different lexical resources

4. A case study – the TOT state

On how easily we talk, sometimes certain words are retrieved with difficulty. This happens when we look for a word in the process of communication and it does not come to mind. This is known as the tip-of-the-tongue state or problem. The TOT phenomenon occurs to each individual at least once per week (Brown, 1991). It is clear that a tool that would help a human to recover a forgotten word during a conversation or during a creative writing process would have to be handy, act fast and involve a very large collection of words properly manipulated.

Zock (2015) proposes a three-step methodology. He says that in search for a word that does not come to mind (target word), we always start from one or more source words. In a first step of the search, out of this source word(s), an extension process

would have to generate a large number of associated words, too large to be handled visually by the subject. Then, in a second step, this family of words would have to be clustered automatically, and containing a manageable number of semantically/phonetically/etc. related words. If the number of clusters is itself manageable, then, in the last step, the subject should decide among the representative words of clusters, which one is the closest to the target word. Once chosen, s/he would have to search the cluster and, if lucky, the target word is there. If unlucky, the process should be iterated starting from a subset of the chosen cluster, which is supposed now to be closer to the target word than at the beginning of this search process.

A strategy spontaneously used by humans to revive a missing word in memory is to access a lexical resource for synonyms, hyponyms or hypernyms of the term that comes in mind first as being close enough to the target.

The TOT states are common for speakers, when engaged in discussions, as well as for writers, who can lose their ideas when searching for the forgotten word. The unpleasant phenomenon makes pressure on persons' mind, because it interrupts the fluency of the dialog. In order to reach the target word they try to make connections to similar words, synonyms, antonyms, etc.

We consider a concrete example: let the target word be *vicar* (En. *vicar*, *dean*), supposed temporarily forgotten, and let the first word that crosses our mind be the noun *abate* (En. *abbot*).

In RoWordNet, the example taken into consideration – *abbot* – is registered in the religious domain. This is a clue for extracting from this resource all words connected with this domain. The same algorithm is applied to the DEX-online resource, selecting all words contained in the dictionary definition of *abbot*. Further, we do a similar selection in the COROLA corpus, but for the neighbours of the entry word in a 5-5 left-right window. The union of all these selections cumulates a list of about 300 words, classified as keywords. In order to ease the process of searching the target word, the list of keywords was sorted alphabetically before being presented to the user. The targeted word *vicar* was recognised there by the user.

5. Conclusions and future work

Each type of lexical resource represents information about language in a specific way. In a trial to complement the specific information, therefore putting these sources together, it is clear that some of the information becomes redundant, while each resource contributes with new information.

In our paper we presented an approach for representing in a unified manner a number of distinct types of lexical resources. The proposal starts from the idea to represent only once the common information and to add to this the specific information as contributed by each resource.

As application for our model, we presented an instance of the TOT problem. The intent was to recuperate a forgotten word starting from a seed word, by using a combination of four types of lexical resources that have been previously brought to a uniform representation.

We intend to continue this research by adopting the RDF (Resource Description Framework) representation formalism to represent linguistic information. Being a graph-based data model (Chiarcos *et al.*, 2012) and laying the representation on the very simple notion of subject-verb-object triples, RDF facilitates data merging for both structured and unstructured data that have to be combined, exposed and shared across various applications.

Acknowledgements

This work was co-funded by the European Social Fund through Sectoral Operational Programme Human Resources Development 2007 – 2013, project number POSDRU/187/1.5/S/155397, project title "Towards a New Generation of Elite Researchers through Doctoral Scolarships."

References

Bibiri, A. D., Bolea, C., Scutelnicu, L. A., Moruz, A., Pistol, I.C., Cristea, D. (2015), Metadata of a Huge Corpus of Contemporary Romanian. Data and organization of the work, in *Proceedings of the 7th Balkan Conference in Informatics*. ACM New York. ISBN 978-1-4503-3335-1.

Brown, A.S. (1991). A Review of the Tip-of-the-Tongue Experience, *Psychological Bulletin*, Southern Methodist University, vol. 109, No. 2, pp. 204-223.

Chiarcos, C., Hellmann, S., Nordhoff, S. (2012). Introduction and Overview, in C. Chiarcos, S. Hellmann, S. Nordhoff (eds.) *Linked Data in Linguistics. Representing and Connecting Language Data and Language Metadata*. Springer-Verlag, Berlin Heidelberg, pp. 1-12.

Fellbaum, C. (ed.) (1998). WordNet: An Electronic Lexical Database, Cambridge, MA, Mit Press.

Gîfu, D. (2010). Utilization of technologies for linguistic processing in an electoral context: Method LIWC-2007 in *Proceedings of the Communication, context, interdisciplinarity Congress*, vol. 1, Ed. "Petru Maior" University, Târgu-Mureș, pp. 87-98.

Sinclair, J. (1994). Corpus Typology, EAGLES DOCUMENT EAG-CWG-IR-2, Version of October.

Teubert, W. (1997). Language Resources for Language Technology, in Dan Tufiș, Poul Andersen (eds.) *Recent Advances in Romanian Language Technology*, Romanian Academy Publishing House, Bucharest, pp. 25-34.

Tufiş, D., Cristea, D. (2002). Methodological Issues in Building the Romanian Wordnet and Consistency Checks in Balkanet. In *Proceedings of LREC*, Las Palmas, pp. 35-41.

Tufiş, D., Barbu Mititelu, V., Ştefănescu, D., Ion. R. (2013). The Romanian Wordnet in a nut-shell. In *Language Resources and Evaluation*, 47(4), pp. 1305-1314.

ALIGNED DEPENDENCY TREEBANK ENGLISH-ROMANIAN-FRENCH

CĂTĂLINA MĂRĂNDUC[1,2], CENEL-AUGUSTO PEREZ[1], RALUCA-ȘTEFANA BALMUȘ[1]

[1] *Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iași*
[2] *"Iorgu Iordan – Al. Rosetti" Institute of Linguistics, Romanian Academy, Bucharest*

catalina.maranduc@info.uaic.ro, augusto.perez@info.uaic.ro,raluca.balmus@info.uaic.ro

Abstract

This paper describes a special sub-corpus of the UAIC-RoDepTb that contains the first 250 sentences from the sub-corpus "1984", buit on the basis of Orwell's novel. We searched the aligned sentences in the English version in the MULTEXT-East project and the same sentences in a French version of this novel. The English and the French versions have been transformed into the format of our TreeAnnotator interface, and then annotated in the same conventions of UAIC-RoDepTb. The Romanian version is automatically parsed and then manually corrected; for the other two versions we had not a parser trained for these languages and for our conventions, so they have been manually annotated with syntactic information. We used the MaltEvalViz to compare first the Romanian and then the French version with the English original version. We comment on the differences between the syntax of these languages.

Keywords: dependency Treebank, comparison of trees, aligned corpus, evaluation of translations.

1. Introduction

This project intends to be useful for the Example Based Machine Translation (EBMT) systems. The comparison of trees can be also useful in the evaluation of machine translations. Romanian language has an insufficient rate of computerization, it has yet no large annotated corpora, which makes this type of translations incomplete and with few optimal results.

It would be preferable to build large corpora and this would be the surest way to improve the performance of MT for Romanian language, but such corpora are built with big costs of time and money. Therefore, in this moment, it is preferable to build rule based tools.

By introducing rules resulting from the comparison of examples enriched with a complex annotation, substantial improvement in MT can be reached for Romanian language. The role of the NLP linguists is to build resources that will help in the development of different programs.

So we have chosen the EBMT model in the most complex version, where the examples are syntactic trees for the source and the target languages, using a parser that analyzes the trees in the source language, and then, after the parsed segments are mapped, it generates trees in the target language.

A practical reason for the choice of this model is that our NLP group has already a resource of the syntactic type, Romanian Dependency Treebank (UAIC-RoDepTb), so we used some sentences from it as the Romanian version and we began to select aligned English and French sentences with those sentences to build a parallel corpus.

Because a sub-corpus of UAIC-RoDepTb uses the Romanian version of Orwell's novel "1984", in the MULTEXT-East[1] project (Erjavec, 2001; Erjavec, 2004), we decided to use the same source to obtain the English original version of the novel, selecting from it about 1000 sentences. But French is not in the 7 languages of this project, so we found a translation of the novel in French and processed it with the TTL POS-tagger[2] for French, on the RACAI site.

2. Challenges

We do not have a parser that makes the operations described above. It could be built either by statistical methods or by the introduction of rules. If we can't yet put together a large corpus, we try to formulate rules for building a rule-based parser. We know that there are parsers trained to English and French, we can find also Dependency parsers, but each Dependency Grammar has its tags and conventions, we had the intention to align trees built in our convention, compatibles with the annotation of our Romanian UAIC Treebank. The parser must be trained on the same set of tags in each language and our 250 sentences are not enough for this purpose.

Romanian is a language with a rich morphology and a free order of words, it differs from the other two languages with international circulation; in Romanian are no rules to establish the place of the subject and the direct objet, so the alignment of words is difficult to be done.

Other shortcomings are the more free translation of the French version and the fact that the TTL POS-tagger annotated the apostrophe as a separate tag, when the UAIC POS-tagger[3], which has not a French version, considers it as part of the word in which it replaces a letter.

[1]http://nl.iis.si/ME
[2]http://www.racai.ro
[3]http://nlptools.info.uaic.ro

3. State of the art

There is an automatic translator with pretty good performance among the RACAI web-services tools. The tool uses both statistical method and rules, but has no parallel corpora in form of a Treebank, that would be useful and can improve performance of the program. In (Colhon, 2012) a study of English - Romanian parallel syntactic trees in the terms of Constituents Grammar is made, also with the purpose to be used in EBMT. A study of the alignment of comparable corpora is made also in (Ion, 2012).

Sanguinetti et al (2014) aims to introduce the issues related to the syntactic alignment of a dependency-based multilingual parallel Treebank, ParTUT. This approach starts from a lexical mapping and then attempts to expand it using dependency relations. In developing the system, however, they realized that the only dependency relations between the individual nodes were not sufficient to overcome some translation divergences, or shifts, especially in the absence of a direct lexical mapping and a different syntactic realization.

For this purpose, they explored the use of a novel syntactic notion introduced as a dependency theoretical framework, i.e. that of catena (Latin for ”chain”), which is intended as a group of words that are continuous with respect to dominance. In relation to the task of aligning parallel dependency structures, catenae can be used to explain and identify those cases of one-to-many or many-to-many correspondences, typical of several translation shifts that cannot be detected by means of direct word-based mappings or bare syntactic relations.

Their paper describes the overall structure of the alignment system as it has been currently designed, how catenae are extracted from the parallel resource, and their potential relevance to the completion of tree alignment in ParTUT sentences.

In another paper (Tiedemann, 2010), the author presents an experimental toolbox for automatic tree-to-tree alignment based on local classification and alignment inference. The aligner implements a recurrent architecture for structural prediction, using history features and a sequential classification procedure. The discriminative base classifier uses a log-linear model which enables simple integration of various features extracted from the data. The Lingua-Align toolbox provides a flexible framework for feature extraction including contextual properties and implements several alignment inference procedures. Various settings and constraints can be controlled via a simple frontend or called from external scripts. Lingua-Align supports different Treebank formats and includes additional tools for conversion and evaluation.

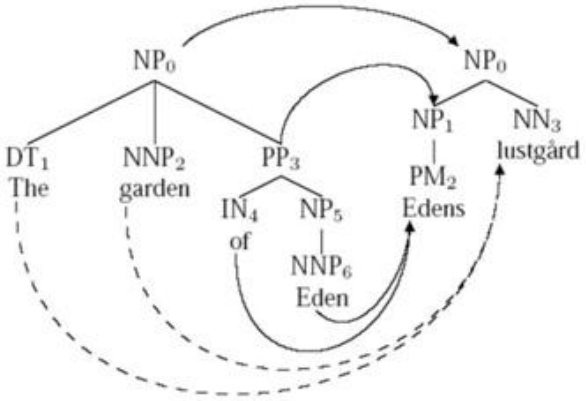


Figure 1: Tree-to-tree alignment in (Tiedemann, 2010).

Their experiments show that this tree aligner produces results with high quality and outperforms unsupervised techniques proposed otherwise. It also integrates well with another existing tool for manual tree alignment which makes it possible to quickly integrate additional training material and to run semi-automatic alignment strategies.

Another paper (Richardson *et al.*, 2014) introduces the Kyoto EBMT framework. Their system uses a tree-to-tree approach, employing syntactic dependency analysis for both source and target languages in an attempt to preserve non-local structure. The effectiveness of their system is maximized with online examples matching and a flexible decoder. Evaluation demonstrates BLEU scores competitive with state-of-the-art SMT systems such as Moses.

Another paper (Sulger *et al.*, 2013) discusses the construction of a parallel Treebank currently involving ten languages from six language families. The Treebank is based on deep LFG (Lexical- Functional Grammar) that were developed within the framework of the ParGram (Parallel Grammar) effort.

This grammar produces output that is maximally parallelized across languages and language families. Its output forms the basis of a parallel Treebank covering a diverse set of phenomena. The Treebank is publicly available via the INESS Treebank environment, which also allows for the alignment of language pairs. They thus present a unique, multi-layered parallel Treebank with alignment of sentences at several levels: dependency structures, constituency structures and POS information.

4. Objectives

4.1. Limitation of our research

The cited papers show us that we are at the beginning of such an enterprise. Although the idea of *catenae* is interesting, we cannot apply it for our aligned Treebank. We will transform the aligned corpus into Universal Dependencies (UD) format and then we will align its words using the label *mwe* (multi word expression) in the case of one-to-many or many-to-one alignments.

The UD project, to which we intend to affiliate our entire Treebank, is a big family of dependency Treebanks in more than 20 languages, which respects the same format and conventions of morphologic and syntactic annotation. So, we can find parallel colpora on this commun system of annotation.

Our small corpus can be the core of a new corpus that can be extended later with more sentences and more languages. It will be made available on our NLP-tools site[4], for the benefit of future MT tasks.

4.2. Collecting the texts

It is possible to collect texts that are already annotated, from other alignment projects of texts in several languages, if we have access to them. The project MULTEXT-East (MTE) for the Orwell *"1984"* novel has aligned Bulgarian, Czech, Estonian, Hungarian, Lithuanian, Russian, Slovenian, Serbian, Romanian versions with the English original version.

4.3 The XML format

In the first stage of our project, the selection of texts, the English version collected from MTE and the French version processed by the TTL POS-tagger, resulted in different XML annotation formats, incompatible with the format of the Romanian Treebank, so we needed to pass through different Perl programs that would bring them to the XML format required by the TreeAnnotator interface (with which we currently work).

In this format the text is split into sentences, then in words (but not in letters), then the words are morphologically analyzed, having two completed fields ”lemma” and ”postag” and having at the end of each word two empty fields to be annotated using the TreeAnnotator interface: ”head” and ”deprel” (dependency relation).

4.4 Manual annotation

In our project, aligning a small number of sentences, we do not use an automatic syntactic annotation for the English and the French version. We have only a version

[4] nlptools.info.uaic.ro

of the MaltParser[5], built by Joachim Nivre (Nivre *et al.*, 2006), and trained on the UAIC-RoDepTb. If we will extend the dimensions of the parallel Treebank, we must find a solution of this problem because there are parsers for all the languages and we have to solve only the problem of different formats. The annotation conventions in which the other parsers work are different than the Romanian Treebank conventions, with which we want to align the English and French versions, so it would require a program for the transposition of different conventions (or to train the parsers from English and French on a big number of sentences annotated in our conventions).

A good solution would be to continue this project after the affiliation of our UAIC-RoDepTb to the UD (Universal Dependencies) project, adopting the international annotation conventions and easily finding annotations for more languages in the same format. However, the automatically parsed Romanian texts have around 70% accuracy for the "heads" and "labels". The statistics are made comparing the automatic annotation with the manually checked version, counting first the number of regents correctly detected for each word, "heads", then counting the number of syntactic relations correctly annotated as "labels" of the relations. To manually correct the frequent errors of the parser sometimes takes no less time than for manual annotation.

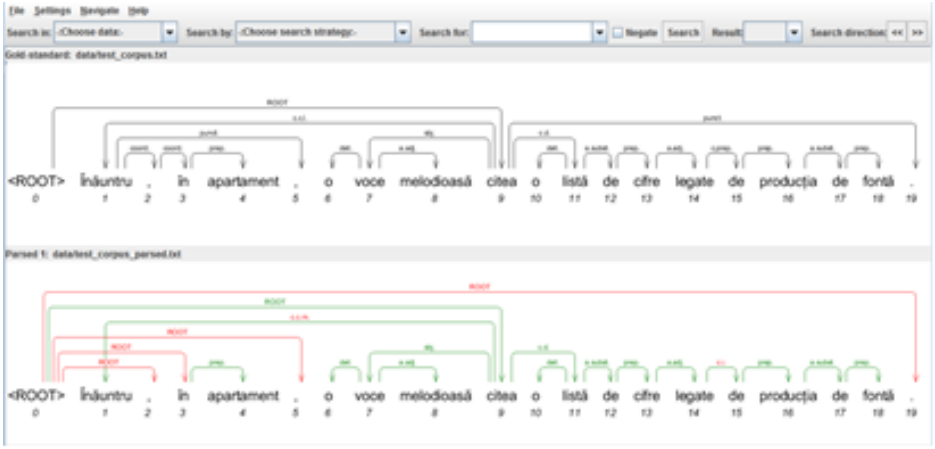


Figure 2. The MaltEvalViz

5. Comparing trees

For this step we needed to build an interface to display on the same page more graphs and a system to be able to compare XML formats, in order to report the differences.

[3] This parser can be found at http://nlptools.infoiasi.ro/WebFdgRo/.

We used the MaltEvalViz6 interface, which can visualize only two trees at a time, being made to compare an automatic annotated tree with the gold version obtained by manual correction. (Fig. 2).

Another possibility to compare the three trees is to open three times our program TreeAnnotator (Fig. 3).

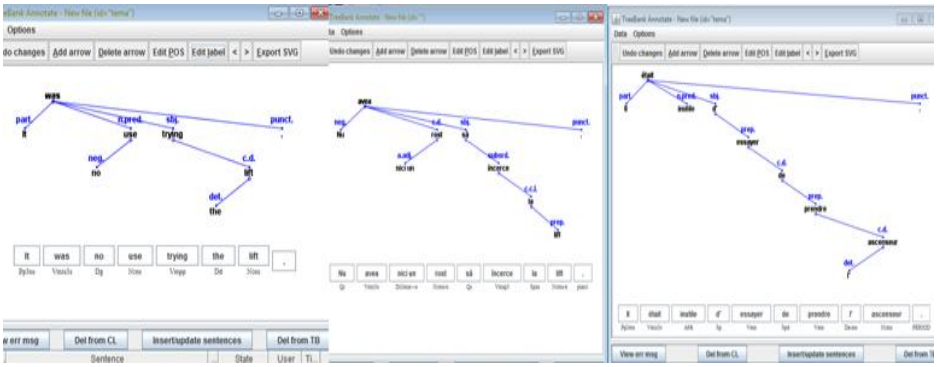


Figure 3. Comparison of trees with the TreeAnnotator.

6. Commenting the results

6.1. Statistical comparison

The alignment of the sentences has the following result: The English version has 250 sentences, the Romanian version has 240 sentences and the French version has 244 sentences.

Table 1. The first 10 lines of the table with the alignment of sentences

1984.en.	1984.ro		1984.fr.	
sent.id	sent.id	word.id	sent.id	word.id
1	1	1-15	1	1-17
2	1	16-64	1	18-75
3	2		2	
4	3	1-14	3	1-25
5	3	15-48	3	26-65
6	4		4	
7	5		5	
8	6	1-21	6	1-21
9	6	22-34	6	22-36
10	7		7	

[6] The program was built by Johan Hall, Jems Nilsson and Joakim Nivre; it is open source and can be found at http: //www. maltparser. org/malteval.html.

In this table we transcribed only the alignment of the first 10 sentences of our corpus. The empty columns in indicate that no translation has been provided for the respective English sentences in neither Romanian nor French.

We did not yet make a word-to-word alignment; we must first rezolve the problem of the erroneous annotation of the apostrophe in the French version. This problem is important because the Dependency Grammar conventions force us to annotate the punctuation as tags in the tree.

We made another statistical interpretation, counting the number of relations of each type and comparing the results in the case of the three languages. We extracted here only the rows with big differences from this statistics (see table 2). The sum of percents is not 100 because the table does not content all the relatons.

Table 2. Differences between the 3 languages concerning the number of relations

	1984.en		1984.ro		1984.fr	
deprel	5036	100%	5052	100%	5755	100%
a.adj.	473	9.39%	391	7.73%	463	8.04%
a.pron.	17	0.33%	35	0.69%	9	0.15%
a.subst.	266	5.28%	278	5.50%	301	5.23%
aux.	189	3.75%	84	1.66%	105	1.82%
c.c.l.	136	2.70%	144	2.85%	121	2.10%
c.c.m.	254	5.04%	314	6.21%	215	3.73%
c.c.t.	101	2%	120	2.37%	91	1.58%
c.d.	208	4.13%	294	5.81%	247	4.29%
c.prep.	117	2.32%	100	1.97%	128	2.22%
comp.	32	0.63%	74	1.46%	41	0.71%
coord.	407	8.08%	494	9.77%	440	7.64%
det.	492	9.76%	237	4.69%	559	9.71%
neg.	38	0.75%	56	1.10%	92	1.59%
part.	117	2.32%	26	0.51%	101	1.75%
prep.	610	12.11%	660	13.06%	743	12.9%
punct.	511	10.14%	615	12.17%	933	16.21%
refl.	6	0.11%	112	2.21%	62	1.07%
sbj.	444	8.81%	248	4.90%	441	7.66%
subord.	160	3.17%	267	5.28%	201	3.49%
head=0	250	-	240	-	244	-
sentences	250	-	240	-	244	-

The bigger number of relation and of punctuation for the French version is caused by the TTL POS-tagger which analyzed the apostrophe as a separate tag. The final

two lines verify that each sentence has a single root. We can observe that Romanian is the language with a smaller number of subjects (4.9%), when English has the bigger percent, 8.81, and French has 7.66%, due to the ellipsis of this relation in Romanian. This language has the bigger number of comparatives and temporal, modal, local adverbials. The comparative and superlative degree is not sinthetical expressed in Romanian (see example 5) The relation comp. has 1.46% occurences in Romanian and only 0.63% in English, 0.71% in French. The English has not reflexives (0.11%, but 2.21 in Romanian, 1.07 in French), the French has a big number of negations, because it doubles the negation (1.59%, but only 0.75% in English and 1.10% in Romanian), etc.

6.2. A note on differences among languages

The observation of differences and the formulation of transposition rules for the translation between the language pairs were made in view of a future MT system.

We sorted the differences type identified seeing the parallel trees and we formulated rules only for the differences that occur with some frequency. The other differences, for which we did not formulate rules, will be solved by the MT program with statistical methods when the corpus will increase in size. By increasing the size, it is also possible that differences which now occur as isolated cases, to show up in a sufficient number of cases as to be considered common differences and become profitable for the formulation of rules.

6.3 Some difficulties

When the splitter segmented the sentences after the (;) or (:), we have manually corrected the error.

The biggest difficulties are produced by the separation of the apostrophe in the French version, and the recognition of the part of speech of the word preceding it. The annotation of the apostrophe as a punctuation element generated noise on statistical interpretation of the number of relations (see table 2).

It is difficult to recognize phrasal verbs of all kinds, especially followed by a preposition and to link this preposition with the verb, which triggered some differences among the corresponding trees (see the small number of prepositional objects in the English version). Instead, the POS-tagger analyzed everywhere, by mistake, two words as a multi word expression. The French translation is more free than the Roumain translation; there should be a greater closeness to the source text.

7. Morphosyntactic structure differences among languages. Some common mistakes

The subject is a function that appears in French more frequently; it is seldom understood and never included.

The three languages have different prepositional regime: The "c.c.t." (temporal modifier) function is without preposition in English and French, but with preposition in Romanian:

Examples:

one day = intr-o zi = un jour. (1)

The genitive is formed with preposition in French and English but without preposition in Romanian:

girl's hand = mâna fetei = la main de la fille. (2)

The partitif in French is formed with preposition and in English and Romanian without mark.

as the acid = ca acidul = comme de l'acide. (3)

In French and in English there are more *c.prep.*(prepositional object) and less *c.d.*(direct object) than in Romanian (see Table 2).

The French and English version have a lot of definite articles; in Romanian the definite article is a suffix at the end of the noun and can't be annotated.

the watches = ceasurile = les horloges. (4)

In the French version, *du, des, au, aux* are annotated like prepositions, but they contain also a definite article which cannot be annotated.

In English and in French there are synthetic comparatives which cannot be annotated and in Romanian there are analytic comparatives; this is the cause for the bigger number of *comp.* relationships in Romanian.

more = mai multe = plusieures (5)

In French there are several auxiliary besides the ones from English and Romanian language: *venait de* = auxiliary for the recent past.

has just taken = tocmai luase =venait de prendre. (6)

Aller = auxiliary for the future.

will undertake = va întreprinde = il allait entreprendre. (7)

The negation is doubled in French, while in English it is formed with an auxiliary.

he did not know = nu ştia = il ne savait pas. (8)

In English exist negative pronouns with a syntactic function instead of a negative

value: *sbj.*(subject) in the following examples).

nobody knew = nimeni nu ştia = personne ne savait. (9)

The French subjunctive is not synonymous with the Romanian conjunctive and it can't be used identically. The Romanian conjunctive must be translated into French and English as an infinitive (without the particle to).

could speak = putea să vorbească = pouvait parler (10)

The French subjunctive must be translated by the Romanian indicative. The following examples show the past time; this is similar for the present.

although he made = cu toate că făcu = bien qu'il fit. (11)

There are a lot of grammatical homonyms that must be disambiguated. "*en"* is sometimes in French the gerundive particle, but sometimes it is a preposition; the same situation in the case of the Romanian supine particle "de".

stopping = oprindu-se = en s'arretant (12)

The French "en" and "y" either are prepositions or indefinite pronouns; in these cases we annotated them as "*c.prep."* (considering that *en=de cela, y=à cela).*"y" is sometimes a particle belonging to a phrase *(il y a = se află),*or it has sense:

he goes there = intrase acolo = y etait entré. (13)

"de" can be a marker of the accusative case (*de*+ undetermined noun in accusative case):

April day = *zi de aprilie = journée d'avril.* (14)

or "de" can be a marker of the genitive case (*de*+ determined noun in genitive case):

the lift-shaft = cutia ascensorului = cage de l'ascenseur. (15)

-l, le, la, les + noun ([1]) = the determiner is homonymous with *l, le, la, les* + verb ([2]) = the accusative pronoun, direct object.

the boy knows him = băiatul îl cunoaşte = le[1] garcon le[2]connait. (16)

"it" (*"il"*) is subject if it is a co-referential character, but it is a particle with impersonal value if nobody is making the action.

it rains = plouă = il pleut. (17)

"C'est" and "*on"* are marks of the impersonal value, it is sometimes convenient to consider them subjects, sometimes they coexist with the real subject, a sentence or a

verbal abstract noun.

Multi word expressions are specific for each language:

although = deşi = bien que. (18)

In this case we annotated "*bien"* as conjunction and "*que"* as particle because it is followed by a subjunctive verb.

he walk away = o luă pe jos = alla, monta. (20)

In Romanian, "*aşa încât"* introduces a consecutive clause and in French its translation is a coordinating conclusive conjunction; we can remark that the subordination and the coordination are not corresponding in the three versions; in Romanian there is the bigger number of coordination relations, and in English there is the smaller number of subordination relations.

In Romanian and in French the adjective is accorded, having the same number and genre with the noun, but this accord does not exist in English. We annotated "*there is"* like a particle of impersonal value and "*itself"* like a reflexive, but it has a very small number of attestations.

"No" is a negation which belongs to the noun; in Romanian all negations belong to verb. *"In", "on"* after the verb and without introducing a noun are not prepositions, but adverbs; if the prepositions belonging to a verb change its meaning, we must annotate it as "*part."* (particle):

get off = a coborî = descendre (21)

In English, the determinants of a noun are all above it, in French more determinants are shared above and below the noun, and in Romanian the below position is preferred. There are many attributes expressed by a noun that didn't have preposition, comparing to French.

We can make rules from these examples by replacing the words occurring with their morpho-syntactic annotation in a MT rule-based system.

This is an example in our annotation system:

stopping = oprindu-se = en s'arretant →

[En] Vmg = [Ro] Vmg Px3--aw = [Fr] Qg Px3--aw Vmg

8. Conclusions and future work

The aligned Treebank can be useful for MT systems and for the evaluation of translations. We described here only the core of such a resource. We must continue to develop it after the affiliation of our Treebank at the Universal Dependencies project.

We intend to continue our research by finding specialists for other languages, German, Russian, Italian, Spanish, and also by increasing the corpus of examples. We will align it not only tree-to-tree but also word-to-word.

References

Colhon, M. (2012) Language Engineering for Syntactic Knowledge Transfer. In *ComSIS Vol. 9, No. 3, Special Issue,* September 2012, 1231-1247.

Erjavec, T. (2001). Harmonized Morphosyntactic Tagging for Seven Languages and Orwell's 1984. In *Proceedings of the 6th Natural Language Processing Pacific Rim Symposium,* Tokyo, p. 487-492.

Erjavec, T. (2004) MULTEXT-East Version 3: Multilingual Morphosyntactic Specifications, Lexicons and Corpora. In *Proceedings of the Fourth Intl. Conf. on Language Resources and Evaluation,* LREC'2004, ELRA.

Ion, R. (2012) PEXACC: A Parallel Sentence Mining Algorithm from Comparable Corpora. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*. Istanbul, Turkey, May 2012

Nivre, J., Hall, J., and Nilsson, J.: (2006). MaltParser: A Data-Driven Parser-Generator for Dependency Parsing. In *Proceedings of the Fifth Intelligence Conference on Language Resources and Evaluation*, LREC, May 2426, 2006, Genoa, Italy, p. 2216-2219.

Richardson, J., Cromières, F., Nakazawa, T., Kurohashi, S. (2014). Kyoto EBMT: An Example-Based Dependency-to-Dependency Translation Framework. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations,* 79-84.

Sanguinetti, M., Bosco, C., Cupi, L. (2014). Exploiting catenae in a parallel treebank alignment. In *Proceedings of LREC 2014,* 1824-1831.

Sulger, S., Butt, M., Holloway King, T, Meurer, P., Laczkô, T., Rákosi, G., Bamba, Dione C., Dyvik, H., Rosén, V., De Smedt, K., Patejuk, A., Çetinoglu, O., Arka, I. W., and Mistica, M. (2013). ParGramBank: The ParGram Parallel Treebank. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, 550-560.

Tiedemann, J. (2010). Lingua-Align: An Experimental Toolbox for Automatic Tree-to-Tree Alignment. in *Proceedings of LREC 2010,* 735-743.

NOUN-VERB DERIVATION IN THE BULGARIAN, ROMANIAN AND ENGLISH WORDNETS – A COMPARATIVE APPROACH

VERGINICA BARBU MITITELU[1], BORISLAV RIZOV[2], EKATERINA TARPOMANOVA[2], SVETLOZARA LESEVA[2], TSVETANA DIMITROVA[2]

[1] *Research Institute for Artificial Intelligence "Mihai Drăgănescu", Romanian Academy, Bucharest*

[2] *Institute for Bulgarian Language, Bulgarian Academy of Sciences, vergi@racai.ro, {boby, katja, zarka, cvetana}@dcl.bas.bg*

Abstract

In the context of developing wordnets and using them in various applications, we have been enriching the Romanian and Bulgarian resources with morphosemantic relations that can aid broadening the wordnet content and improving the possible NLP applications. In this paper, we build on our previous results, adding to our presentation data from English. As a consequence, we offer a comparative study on noun-verb derivation in the three languages (from three different branches of the Indo-European language family). Our results can serve as training material for the automatic identification and assignment of derivational and morphosemantic relations needed in various applications.

Keywords: wordnet, derivation, affixes, morphosemantic relation, Bulgarian, English, Romanian.

1. Introduction

This paper presents the results of the harmonised enrichment of the Bulgarian and the Romanian WordNet with morphosemantic relations. Some partial results of our work were presented in Tarpomanova *et al.* (2014). In this paper we add one more data set (i.e., derivational and morphosemantic relations in the English wordnet) in the presentation and discussion of linguistic facts. In the future we intend to propose a comprehensive multilingual analysis based not only on noun-verb pairs but also on pairs from other parts of speech (adjectives and adverbs) as well.

Our interest in morphosemantic relations between wordnet synsets finds its place within a larger group of wordnet developers focusing on such relations (Bilgin, 2004; Pala, 2007; Koeva, 2008; Piasecki, 2009; Sojat and Srebacic, 2014). Depending on the derivational specificities of the language and/or the methodology adopted, different, possibly overlapping sets of morphosemantic relations have been identified and implemented in different wordnets.

In the Princeton WordNet (PWN) (Fellbaum, 2009), morphosemantic relations are encoded as a stand-off file. These semantic links are established between literals "that are similar in meaning and where one word is derived from the other by means of a morphological affix" (Fellbaum, 2009).

The PWN morphosemantic links were automatically transferred to the Bulgarian and the Romanian WordNet provided that both synsets that were members of a relation, were implemented in those languages. Afterwards, the teams working on the two wordnets performed automatic extraction of derivationally related literal pairs and derivational models from the morphosemantically related synsets, followed by manual validation of the pair members.

The goal of this paper is to summarise the findings of our joint work with a view to proposing (i) a more comprehensive account of the derivation in the studied languages, as well as (ii) a framework for automatic discovery of derivational relations and automatic assignment of morphosemantic relations which makes use of the rich inventory of derivational patterns of the languages under study. Some of the efforts already undertaken along these lines are mentioned throughout the paper. A further objective is to implement these linguistic generalisations in applications that benefit from these relations.

2. Derivational morphology of Bulgarian, Romanian and English

The languages under focus belong to three different branches of the Indo-European language family – Slavic (Bulgarian), Romance (Romanian) and Germanic (English). However, some similarities can be found in their derivational morphology, as we will show below.

In Bulgarian and Romanian suffixation is the most productive means of word formation, but also the most complicated one: one or more suffixes may be added to a stem, or a suffix may be substituted for another; suffixation may or may not change the part of speech of a word, while prefixation (usually) does not change the part of speech. In Bulgarian, prefixes have an important role for verb-to-verb derivation as they may involve change of verbal aspect. When the two derivation processes, suffixation and prefixation, occur simultaneously to form a new word we talk about parasynthetic derivation. In English, suffixation contributes to forming nouns from verbs and verbs from nouns in a great measure. Prefixation and parasynthetic derivation cannot be found in the English data we worked with.

Conversion is a disputable notion in the traditional linguistic descriptions of both Romanian and Bulgarian. According to the Romanian tradition, it is distinct from derivation and always implies homonymy. In the Bulgarian literature, conversion is usually interpreted in a broader sense as a process of word formation in which the written forms of two words in a derivational pair differ only by their inflectional

markers. Formation of deverbal nouns by removing the thematic vowel and the inflection of a verb without adding a suffix to the noun is called zero suffixation. In English, conversion is also called zero-derivation and the two word forms are identical most of the times. This is a very frequent means of creating new words in English, as suggested by the presented data.

In the Bulgarian data discussed below, zero suffixation is subsumed under conversion and labelled accordingly. Cases of conversion in Romanian are not discussed in this paper, as it does not serve to create verbs (in their infinitive form) from nouns or, vice versa, nouns from infinitives.

3. The nature of morphosemantic relations

The morphosemantic relations encoded between nouns and verbs usually denote a relation between a predicate and a participant in its semantic representation: Agent, State, Result, Undergoer, Property, Vehicle, Destination, Material, Body-part, Cause, Instrument, Location, By-means-of (Fellbaum, 2009). The only exception is the relation Event, which links verbs to deverbal nouns denoting the same event.

The semantic label associated with such a relation holds between synsets and is transferable across languages, even though the morphologic relation (between literals) needs not be expressed (Koeva, 2008). Moreover, the semantic relation may be derivationally expressed only by certain literal pairs in the respective synsets: e.g., in the synsets {write, compose, pen, indite} – {writer, author} only 'write' and 'writer' are derivationally related, although each literal in the former synset is semantically related to each literal in the latter synset.

4. Adding morphosemantic relations between synsets in PWN, BulNet and RoWN

The morphosemantic relations holding between English nouns and verbs are not marked as such in the PWN; they can be found in a stand-off file: for each noun-verb pair, the synsets to which they belong are registered, as well as the semantic relation between the two lexemes. These relations were transferred between the corresponding lexicalised synsets in the RoWN and BulNet and the synsets were checked automatically for derivationally related literals. For Bulgarian each candidate pair of such literals was compared using string similarity and heuristics as described in (Dimitrova *et al.*, 2014). For Romanian, the combinations of all single word literals and either a prefix or a suffix (identified in advance) were matched against the list of literals (also extracted in advance) (Mititelu, 2012). After the derivational candidates were identified, they were validated by experts.

The Princeton stand-off file contains 17,739 pairs of morphosemantically related nouns and verbs: more than half of these pairs (8,968) are the result of conversion, while the other 8,771 reflect suffixation. In Romanian 2,767 pairs have been validated so far, with the following distribution: 2,429 cases of suffixation, 318 cases of verbal suffixation, and 20 cases of parasynthetic derivation. In Bulgarian 7,078 pairs were found involving: 5,335 cases of suffixation (noun suffixation – 4,811; verb suffixation – 524), 1085 cases of substitution of a noun suffix for a verb suffix or vice versa, 499 cases of conversion, 134 cases of parasynthetic derivation, 25 cases of non-transparent derivation.

5. Expression of morphosemantic relations through derivational patterns

The table below shows the derivational patterns associated with each morphosemantic relation in Bulgarian (Bg), Romanian (Ro) and English (En), with the number of occurrences found in the respective database (given in brackets). "Total" includes the number of derivational patterns for the respective semantic label.

Table 1: Derivational affixes in Bg, Ro and En wordnets associated with semantic labels.

Semantic label	Bg affixes (no. of occurrences)	Ro affixes (no. of occurrences)	En affixes (no. of occurrences)
Agent **Total** Bg: 38 Ro: 30 En: 11	**noun suffix:** -tel (197), -(y)ach (143), -(n)ik (103), -sht (93), -tor (62), -(y)or (46), -ets (38), -ist (24), -ant/-ent (18), -(y)ar (15), -l (14), -dzhiya/-chiya (13), -er (9), -chik (8), -nie (7), -in (6), -n (6), -ko (6), -tsiya (4), -stvo (4), etc.; **verb suffix:** (56); **conversion:** (11); **parasynthetic:** (2)	**noun suffix:** -(ă)tor (176), -t/-s (31), -re (7), -ant (7), -ar (5), -or (5), -ăreț (5), -aș (5), -(ă)toare(4), -ier (3), -[ăi]cios(3), etc.; **verb suffix:** (35); **parasynthetic:** (2)	**noun suffix:** -er (2279), -or (306), -ation (10), -ion (9), -eer (4), -ence (3), -ee (1) **verb suffix:** (21) **conversion:** (410)
Body-part **Total:** Bg: 7 Ro: 1 En: 4	**noun suffix:** -ka (1), -nie (1), -tel (1), -(y)ach (1), -tor (1); **verb suffix:** (1); **conversion:** (4)	**noun suffix:** -; **verb suffix**: (1)	**noun suffix:** -er (16), -or (11) **verb suffix:** (2) **conversion:** (14)

By-means-of **Total** Bg: 36 Ro: 24 En: 12	**noun suffix:** -ne (66), -nie (54), -ka (32), -tsiya (24), -tor (9), -tel (9), -iya (9), -(n)ost/-est (7), -no (6), -alo/-ilo (5), -ovka (5), -ina (4), -ie (4), etc.; **verb suffix:** (48); **parasynthetic**: (8); **deriv:** (4); **conversion:** (51)	**noun suffix:** -re (98), -(ă)tor (10), -ație (8), -t/-s (6), -(ă)tură (6), -eală (4), -(ă)toare (4), -or (3), -ment (3), etc.; **verb suffix:** – (66); p**arasynthetic:** (6)	**noun suffix:** -er (145), -ation (132), -ion (77), -ment (48), -ance (18), -or (11), -tion (7), etc.; **verb suffix:** (40) **conversion:** (791)
Destination **Total** Bg: 2 Ro: 3 En: 3	**noun suffix:** -at (2) **verb suffix:** (2)	**noun suffix:** -ar (1), -ant (1) **verb suffix:** (1)	**noun suffix:** -ee (13) **verb suffix:** (2) **conversion:** (2)
Event **Total** Bg: 55 Ro: 34 En: 16	**noun suffix:** -ne (2698), -nie (390), -tsiya (366), -ka (93), -stvo (55), -ie (48), -(n)ost/-est (31), -ăk (22), -ezh (21), -iya (18), -(n)itsa (17), -ba (16), -ovka (13), -ek (8), -nya (7), -azh (6), -no (6), -tva (5), -entsiya (4), -itba (4), -(n)ik (4), -ina (4), -at (3), -ar (3), etc.; **verb suffix**: (177); **parasynthetic:** (11); **conversion:** (278); **deriv:** (12)	**noun suffix:** -re (1174), -t/-s (112), -ație (110), -(ă)tură (48), -(e)ală (44), -ment (9), -ie (8), -et (5), -e (4), -[a/e/i]nță (4), -ător/ătoare (4), -aj (3), -(ă)ciune (3), etc. **verb suffix:** (138)	**noun suffix:** -ion (2021), -ation (1467), -ment (388), -ence (111), -ance (97), -al (96), -tion (51), -er (23), -ing (5), -or (4), -ee (2), -eer (1) **verb suffix:** (35) **conversion:** (3847)
Instrument **Total** Bg: 24 Ro: 13 En: 6	**noun suffix:** -tor (19), -tel (15), -er (11), -ka (6), -(y)ach (4), -(y)or (3), -lka (3), -nie (3), etc.; **verb suffix:** (17); **parasynthetic:** (2); **conversion:** (6)	**noun suffix:** -(ă)tor (21), -(ă/i)toare (6), etc.; **verb suffix:** (16); **parasynthetic:** (2)	**noun suffix:** -er (345), -or (62), -ion (1), -ment (1) **verb suffix:** (13) **conversion:** (391)
Location **Total** Bg: 15 Ro: 4 En: 7	**noun suffix:** -ishte (6), -ne (2), -(n)itsa (1), -ing (1), -ka (1), -alo/-ilo (2), -lka (1), -nie (1), -tsiya (1); **verb suffix:** (15); **parasynthetic:** (14)	**noun suffix:** -re (6), -ment (1) **verb suffix:** (1) **parasynthetic:** (1)	**noun suffix:** -ion (14), -ation (9), -ment (8), -er (10), -tion (1) **verb suffix:** (7) **conversion:** (239)

Material **Total** **Bg: 21** **Ro: 7** **En: 8**	**noun suffix:** -tel (18), -tor (15), -tsiya (6), -ant/-ent (5), -ka (2), -lka (2), -nie (2), -at (2), etc.; **verb suffix:** (17); **parasynthetic:**(4); **conversion:** (3)	**noun suffix:** -ant (1), -(ă)tor (3), -tură (1) **verb suffix:** (9)	**noun suffix:** -er (53), -or (8), -ion (2), -ation (1) **verb suffix:** (14); **conversion:** (36)
Property **Total** Bg: 16 Ro: 6 En: 9	**noun suffix:** -nie (31), -(n)ost/-est (24), -ne (15), -tsiya (4), -ie (3), -ba (3), etc.; **verb suffix:** (8); **parasynthetic:** (5); **conversion:** (12)	**noun suffix:** -re (32), -ment (2); **verb suffix:** (10); **parasynthetic:** (1)	**noun suffix:** -ion (33), -ation (28), -ence (26), -ment (12), -ance (3), -tion (1) **verb suffix:** (16); **conversion**: (199)
Result **Total** Bg: 47 Ro: 18 En: 13	**noun suffix:** -ne (61), -nie (38), -tsiya (37), -(n)ost/-est (27), -ka (18), -no (12), -at (7), -ets (2), -iya (3), -ie (3), -tor (3), etc.; **verb suffix:** (80); **parasynthetic:** (28); **conversion:** (41); **deriv:** (1)	**noun suffix:** -re (83), -t/-s (11), -tură (10), -eală (4), -ment (3), -et (2), etc.; **verb suffix:** (66); **parasynthetic:** (3)	**noun suffix:** -ation (198), -ion (123), -ment (36), -er (13), -al (13), -ance (8), -tion (4), -ence (1), -or (1) **verb suffix:** (133); **conversion:** (909)
State **Total** Bg: 23 Ro: 11 En: 10	**noun suffix:** -ne (72), -nie (52), -(n)ost/-est (37), -tsiya (16), -ie (7), -stvo (5), -iya (2), -ika (2), -ota (1), -ăk (1), -ina (1), -ovka (2); **verb suffix:** (13); **parasynthetic:** (5); **conversion:** (19); **deriv:** (5)	**noun suffix:** -re (94), -(e)ală (3), etc.; **verb suffix:** (7); **parasynthetic:** (3)	**noun suffix:** -ion (157), -ation (90), -ment (65), -ence (8), -ance (8), -al (6), -tion (1); **verb suffix:** (13); **conversion:** (180)
Undergoer **Total** Bg: 45 Ro: 10 En: 12	**noun suffix:** -ne (31), -nie (27), -n (12), -ka (11), -at (10), -tsiya (9), -(n)ost/-est (9), -(n)ik (8), -ets (6), -iya (6), -ba (5), -ie (5), -ina (4), -sht (4), etc.; **verb suffix:** (35); **parasynthetic:** (9); **conversion: (**34); **deriv:** (3)	**noun suffix:** -re (27), -t/-s (15), etc.; **verb suffix:** (32)	**noun suffix:** -ation (57), -ee (48), -ion (35), -ment (24), -er (21), -ance (8), -tion (3), -al (1), -ence (1) **verb suffix:** (13) **conversion:** (667)

Uses **Total** Bg: 31 Ro: 14 En: 13	**noun suffix:** -ne (26), -nie (22), -tsiya (12), -ka (12), -stvo (6), -ovka (4), -at (4), -lo (5), -tel (3), etc.; **verb suffix:** (53); **parasynthetic:** (18); **conversion:** (30)	**noun suffix:** -re (24), -(e)ală (3), -ment (3), etc.; **verb suffix:** (34) **parasynthetic:** (2)	**noun suffix:** -ance (37), -ment (5), -ation (21), -ier (2), etc.; **verb suffix:** (12); **parasynthetic:** (23); **conversion:** (659)
Vehicle **Total** Bg: 4 Ro: 3 En: 3	**noun suffix:** -tel (2), -ovach (1), -(y)ach (1), -er (1)	**noun suffix:** -or (1), -er (1); **verb suffix:** (1)	**noun suffix:** -er (31), -or (2); **conversion:** (53)

The statistics (see Table 1) shows that more affixes are found in the Bulgarian data – 269 noun suffixes (in each of their senses), 49 verbal ones, and 13 cases of conversion. In Romanian there are 91 noun suffixes and 45 verbal ones (plus 26 cases of verbal derivation that are equivalent to conversion in Bulgarian). In English one can notice the huge number of conversions and a small number of affixes: 15 unique ones: 12 nominal and 3 verbal.

For Bulgarian and Romanian, the difference in the number of the suffix senses can be explained by the specifics of the derivational morphology of the two languages. As a Slavic language, Bulgarian has a rich inventory of noun suffixes that considerably outnumber the corresponding Romanian suffixes: compare the three most productive Bulgarian suffixes with a primary agentive reading (-tel, -(y)ach, -(n)ik) vs. one such suffix in Romanian (-(ă)tor). Bulgarian has also adopted many Romance (Latin) suffixes through the active borrowing of Romance words (directly or via English), so that the Romanian -(ă)tor has an exact equivalent in Bulgarian -tor; -tsiya corresponds to -ație, and -ant/-ent to -ant. These suffixes have their correspondences in English: -or, -(a)(t)ion, -ant (-ant is not found in the data).

The verbal aspect in Bulgarian is another reason for the greater diversity of patterns as both imperfective and perfective stems may be productive in verb-noun derivation and some noun suffixes may attach preferentially or exclusively to either an imperfective or a perfective verb stem, giving rise to different derived words: both -ne, which combines with imperfective stems, and -nie, which usually selects perfective stems, correspond to -re in Romanian.

Despite the difference in the number of suffixes, the three languages show similarity in the derivational productivity of the morphosemantic relations. The relation with the highest diversity of derivational patterns is Event (55 patterns in Bulgarian, 34 in Romanian, and 16 in English). The next highest diversity is displayed by Agent in

Romanian (expressed by 30 derivational patterns), and by Result in Bulgarian and English (47 and 12 derivational patterns, respectively), but Agent is also very productive with 38 Bulgarian and 11 English derivational patterns.

The greater number of occurrences in Bulgarian and Romanian is correlated with the richness of the derivational patterns; thus, the most numerous relations are Event and Agent, which are also the most frequent in the stand-off file for English (so here these two aspects do not correlate).

The relations with the smallest number of occurrences and derivational patterns are Location, Destination, Body-part, and Vehicle in Romanian and Bulgarian, and Instrument and Material are added to this list for English.

6. Polysemy of affixes

The data we have analysed show great discrepancies between English, on the one hand, and Bulgarian and Romanian, on the other, as far as the polysemy of affixes is concerned (see Table 2). There is a large number of monosemous affixes in the two Balkan languages: 35 for Bulgarian and 45 for Romanian, associated mostly with the labels Event (18 in Romanian: -erie, -anță, -aj, etc., and 7 in Bulgarian: -ulka, -tba, -otevitsa, etc.), and Agent (18 in Romanian: -aci, -angiu, -nic, etc., and 13 in Bulgarian: -chik, -ar/-yar, -chiya/-dhziya, -in, etc.); in English we could not find monosemous affixes. Several other relations – Material, Result, Undergoer, Property, State and Instrument in Bulgarian, and By-means-of, Instrument, State, Result and Vehicle in Romanian are represented by one or a couple of unambiguous suffixes. There is no such case in English. An explanation for this can be the fact that for Bulgarian and Romanian the analysis of all noun-verb pairs is not complete.

Table 2. Polysemy of affixes in Bulgarian, Romanian and English

No. of senses	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
No. of Bg affixes	32	19	6	7	6	4	3	2	2	-	2	1	1
No. of Ro affixes	45	7	5	2	4	4	1	3	1	2	-	-	-
No. of En affixes	-	2	1	-	2	3	-	2	3	1	1	-	-

Polysemous suffixes are usually associated with clusters of relations with one of them being the default reading (estimated in terms of number of instances). For example, suffixes which primarily express the relation Agent can also express relations denoting inanimate agents and causes, such as Instrument, Material, By-means-of. The relations Vehicle and Body-part typically should also be included in this group, but the number of instances is too small so we defer judgement. The relation Uses, which denotes a function or a purpose, is also often expressed by agentive suffixes. In certain cases the same suffix denotes both Agent and Undergoer depending on whether the verb is unergative or unaccusative (Fellbaum *et al.* 2009).

In Table 3 below, we show the two most frequent Bulgarian, Romanian and English suffixes primarily associated with the relations Agent and Event and their other senses expressed by the respective relations. For Romanian we mentioned only one Agent suffix (-tor, but also counted its feminine, -toare, in the statistics). All other Romanian Agent suffixes are far less productive than this one and we disregarded them for this table.

Table 3. Polysemy of the most frequent Event and Agent suffixes in Bg, Ro and En.

Language	**Suffix**	**Default semantic value (no. of occurrences)**	**Other semantic values (no. of occurrences)**
Bg	-ne	Event (2372)	State (68), By-means-of (64), Result (46), Undergoer (28), Uses (25), Property (15), Agent (13), Location (2)
Bg	-nie	Event (353)	By-means-of (53), State (47), Result (32), Property (28), Undergoer (23), Uses (20), Agent (6), Instrument (2), Material (2), Body-part (1)
Ro	-re	Event (1173)	By-means-of (98), State (94), Result (84), Property (32), Undergoer (27), Uses (24), Agent (7), Location (6), Instrument (1)
Ro	-ți(un)e	Event (111)	By-means-of (8), Agent (3), Undergoer (1), Result (1), Uses (1)
En	-ion	Event (2021)	State (157), Result (123), Undergoer (35), Property (33), Location (14), Uses (14), Agent (9), Material (2), Instrument (1),

En	-ation	Event (1467)	Result (198), State (90), Undergoer (57), Property (28), Uses (19), Agent (10), Location (9), Material (1)
Bg	-tel	Agent (169)	Material (17), Instrument (13), By-means-of (6), Undergoer (1), Uses (1)
Bg	-tor	Agent (42)	Instrument (15), Material (12), By-means-of (8), Result (3), Uses (1)
Ro	-tor	Agent (180)	Instrument (27), By-means-of (14), Event (3), Material (3), Uses (1)
En	-er	Agent (2279)	Instrument (345), By-means-of (145), Material (53), Vehicle (33), Event (23), Undergoer (21), Body-part (16), Result (13), Location (10), Uses (2)
En	-or	Agent (306)	Instrument (62), Body-part (11), By-means-of (11), Material (8), Event (4), Vehicle (1), Result (1)

7. Conclusion and future work

We have presented above a comparison of derivationally and semantically related pairs of noun-verb literals in the wordnets of three languages: English, Bulgarian, and Romanian.

One of the applications of these results is in the undertaking of comparative research on the derivation of Romanian, Bulgarian and English, earlier results of which were reported in Tarpomanova *et al.* (2014).

We plan to expand our work by further identifying derivationally related literals and semantically related synsets that have not been discovered so far due to imperfections in the recognition algorithms or because the derivationally related pairs are not morphologically related in English. Along these lines, a method for automatic identification and classification of morphosemantic relations in Bulgarian is reported in Koeva *et al.* (2016). We also envisage to undertake the extension of the analysis to pairs of words involving other parts of speech in wordnets, namely adjectives and adverbs.

Adding morphosemantic relations to wordnets helps to increase the connectivity of their synsets, on the one hand, and to establish procedures for semi-automatic expansion with new synsets, on the other. In this vein, we envisage to use the derivational patterns and other correspondences to automatically generate missing literals in existing synsets. For example, given that Bulgarian nouns with the

suffixes -ne and -tsiya occur frequently in one synset (being synonymous in many cases), we can supplement automatically the relevant nominal synsets if one of the variants is missing. Considering the information that such nouns are eventive/resultative nominalisations, we can automatically generate synsets denoting such nominalisations. The English equivalents to -ne, -tsiya and -re nouns are frequently -ing forms, but in the Princeton WordNet -ing nominalisations are not encoded. The endeavour of automatically enriching a Romanian lexical resource was reported by Petic (2010). With access to the semantic types of the words involved in the derivational pairs, we can expect better results.

From the applications perspective, marking morphosemantic (and derivational) relations explicitly in individual and aligned wordnets can prove useful in text processing and information retrieval both in monolingual and in multilingual context. First of all, morphosemantic relations may serve to identify the semantic roles of NPs in unrestricted text and thus identify the participants in an event. On the other hand, derivationally related words from the same or different parts of speech in short segments of texts often signal fragments of semantically related information and the discovery of such relations may be used to identify intra- as well as intertextual semantic links.

8. Acknowledgements

Part of the work reported in this paper was carried out within the joint project "Enhanced Knowledge Bases for Bulgarian and Romanian" of the Institute for Bulgarian Language, Bulgarian Academy of Sciences, and the Research Institute for Artificial Intelligence "Mihai Drăgănescu", Romanian Academy.

References

Barbu Mititelu, V. (2012). Adding Morpho-semantic Relations to the Romanian Wordnet. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC 2012)*, 23-25 May, Istanbul, Turkey, pp. 2596–2601.

Bilgin, O., Cetinoglu, O., and Oflazer, K. (2004). Morphosemantic Relations in and across Wordnets – A Study Based on Turkish. In *Proceedings of the Second Global Wordnet Conference (GWC 2004)*, 20-23 January, Brno, Czech Republic, pp. 60–66.

Dimitrova, T., Tarpomanova, E., and Rizov, B. (2014). Coping with Derivation in the Bulgarian WordNet. In *Proceedings of the Seventh Global Wordnet Conference (GWC 2014)*, 25-29 January, Tartu, Estonia, pp. 109–117.

Fellbaum, C., Osherson, A., and Clark, P. (2009). Putting Semantics into WordNet's "Morphosemantic" Links. In *Proceedings of the Third Language and Technology Conference*, Poznan, Poland. [Reprinted in: Responding to Information Society Challenges: New Advances in Human Language Technologies. Springer Lecture Notes in Informatics], vol. 5603, pp. 350–358.

Koeva, S. (2008). Derivational and Morphosemantic Relations in Bulgarian Wordnet. *Intelligent Information Systems*, pp. 359–368.

Koeva, S., Leseva, S., Stoyanova, I., Dimitrova, T., Todorova, M. (2016). Automatic Prediction of Morphosemantic Relations. In *Proceedings of the Eighth Global Wordnet Conference (GWC 2016)*, 27-30 January, Bucharest, Romania. (forthcoming).

Pala, K. and Hlavackova, D. (2007). Derivational Relations in Czech Wordnet. In *Proceedings of the Workshop on Balto-Slavonic Natural Language Processing*, pp. 75–81.

Petic, M. (2010). Mecanismele generative ale morfologiei derivaţionale. In *Lucrările conferinței Resurse lingvistice și instrumente pentru prelucrarea limbii române*, București, 6-7 mai 2010, pp. 195-203.

Piasecki, M., Szpakowicz, S., and Broda, B. (2009). A Wordnet from the Ground up. Wroclaw. Oficyna Wydawnicza Politechniki Wroclawskiej.

Sojat, K., Srebacic, M. (2014) Morphosemantic relations between verbs in Croatian WordNet. In *Proceedings of the Seventh Global Wordnet Conference (GWC 2014)*, 25-29 January, Tartu, Estonia, pp. 262–267.

Tarpomanova, E., Leseva, S., Todorova, M., Dimitrova, T., Rizov, B, Barbu Mititelu, V, Irimia, E. (2014) Noun-Verb Derivation in the Bulgarian and the Romanian WordNet – A Comparative Approach. In *Proceedings of the First International Conference Computational Linguistics in Bulgaria*, 4 September, Sofia, Bulgaria, pp. 23-31.

CHAPTER 3
SEMANTICS

CORPUS OF ENTITIES AND SEMANTIC RELATIONS WITH APPLICATION IN GEOGRAPHICAL DOMAINS

[1] DANIELA GÎFU, [1] IONUȚ PISTOL, [1,2] DAN CRISTEA

[1] *Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iași*
[2] *Institute of Computer Science, Romanian Academy, Iași branch*

{daniela.gifu, ipistol, dcristea}@info.uaic.ro

Abstract

Geographical entities and relations found in text can offer significant clues towards understanding the content of the text and its connections to the real world. This paper describes annotation conventions for entities and a number of types of semantic relations, mainly within the geographical domain. The effort is in the direction of development of a new technology with a high impact in education and tourism, which is intended to work as a medium for enhancing e-books (especially geography manuals and travel guides) in which geographical entities are marked and linked to references inside and outside of the text. In this context, three types of semantic relations are considered: referential, structural and spatial. Describing the annotation process and the impact of the added metadata on relevant applications is the main focus of this paper.

Keywords: geographic data, interactive book, annotation conventions, semantic relations, statistics.

1. Introduction

Geographical entities in texts can be defined as one or more (contiguous) words (usually forming a Noun Phrase or part of one) denoting a real-world geographical location (*Romania*, *London*, *Mount Everest*) or related features (distance, geographical surface, the height of a mountain pick). In fact, geographical entities are words or phrases which identify concrete values of geographical features: names of locations, coordinates, rocks, different types of landform, spatial dimensions, natural resources, etc.

Building the entity gazetteer and defining the format of the annotated corpora are essential elements for the work described in this paper. There are many works that introduce different annotated methods, using supervised learning and an indirect supervision technique (Speriosu and Baldridge, 2013), a gazetteer independent method by using density estimation techniques (DeLozier *et al.*, 2015), a reference corpus as the TR-CoNLL (Leidner, 2007) or LGL (Local-Global Lexicon) (Lieberman *et al.*, 2010) corpus, an annotated location information to text, by annotating both the location and facility entities, using ACE corpus (conversations and news magazines) (Mani *et al*., 2010), an annotation toponyms from Twitter

messages (Zhang and Gelernter, 2014), and so on. The Spatial Data Transfer Standard (SDTS[1]) describes a method of representation of real world entities on conceptual level, meaning the entities and their topological and geometric features, also, the relations between two geographical poles or two entities.

The aim of this paper is to present the annotation conventions for geographical entities and semantic relations between them, which supports the definition of patterns containing lexical and syntactic level information which could be used to automatically discover such relations. Annotation work so far has been done on a Romanian 8-th grade Geography textbook, but the process can be generalized to other texts covering similar topics: travel guides, atlases, history and geography textbooks.

The paper is structured as follows: Section 2 briefly describes the background related to annotating semantic relations, Section 3 discusses the annotation process with reference to the semantic relations conventions, Section 4 presents some results and statistics and finally, Section 5 includes the conclusions and directions for future work.

2. Background

In the scientific arena of the computational linguistics and linguistic resources, the end of the 90s shows more and more interest towards the issue of semantic relations. For many users the interest for spatial relations is increasing, when they query geographic information systems (GISs)[2]. The computational model by Shariff, Egenhofer and Mark (Egenhofer & Mark, 1995; Shariff *et al.*, 1998) describes natural language spatial relations via formal spatial relations. If they found a simple division of relations in topological and metric, today, the researchers have diversified this semantic classification, rather drifting away from the geometric characterisation. Examples are the semantic relations classified in spatial, temporal and spatio-temporal [Pereira, 2002].

Starting with the NAACL 2003[3], a new community of NLP researchers and engineers focused on different aspects of the geographic text analysis task. In the literature the researchers have focused mainly on spatial relations, without neglecting the referential relations.

The geographical relations classification was diversified. For instance, Klien and Lutz, (2005) or Roberts *et al.* (2013) proposed topological, distance and direction geographical relations, such as: *in*, *same*, *contains*, *r_contains*, *overleaps*, *near*, *different*, etc. using them as a bridge between real world events which have spatio-temporal properties.

[1] In 1992, SDTS was ratified by the National Institute of Standards and Technology (NIST) as a Federal Information Processing Standard (FIPS 173).

[2] Note, GISs occurs primarily through structured query languages (Egenhofer and Herring, 1993)

[3] http://www.kornai.com/NAACL/WS9/orig.html

In order to define spatial relations between geographical entities, first, these entities must be spatially conceptualized. A spatial relation can be determined by comparing the spatial limits of two geographical pols (Roberts and Harabagiu, 2012; Roberts *et al.*, 2013). Moreover, a geographical corpus was built, and it consists of 162 newswire documents (containing 1,695 spatial relations), a subset of the SpatialML corpus (Mani *et al*., 2008). SpatialML is an annotation scheme used to mark the locations in natural language, covering proper names and coreference, relative and absolute positions, plus a set of spatial relations between entities. They proposed the following relations: *in* (for inclusion), *ec* (extended connection), *nr* (near by), *dc* (discrete connection), *po* (partial overlapping), and *eq* (equality). To improve F-Measure, the authors applied a disambiguation method for the similar locations, but which designates separate entities.

For semantic annotation of geo-spatial data, Klien and Lutz (2005) developed tools for specifying queries semantically. Blessing and Schütze (2010) introduced self-annotation, focusing on German entities found in Wikipedia that allows users to eliminate manual labelling. Also, Blessing and Schütze concentrated on geospatial entities on a fine-grained level, although approach is applicable to other domains as well. They followed a supervised extraction approach, considering several features on different linguistic levels.

Shariff *et al.* (1998) suggested a typology with 19 spatial relations that can be detected based on a model including 9 possible intersections between any two geographical entities. The authors have proposed a series of 15 metric relations, as well, aiming to emphasize the idea of distance.

Through this study we were interested to inventorize the quasi-totality of the geographical relations expressed in natural language, hoping also to bring new elements in their classification that will facilitate their automatic recognition in free texts that deal with geographical topics. We adopted a more functional approach in defining semantic relations, in direct connection with the objectives of our MappingBooks project. The idea of the project is to evidence (based on text clues only) in a graphical way, on an attached map, those relations that are fit for such a representation. In this way, the maps themselves will mix information found in outside sources with that explicitly expressed in the text under focus.

3. Annotation process and semantic relations conventions

In this section we will describe the entire annotation process, including the annotation methodology that has many similarities to the one in SpaceML (Mani *et al*., 2008). In identifying the nature of relations we recognised the importance of some clue words or group of words, which we have called *triggers*, and which connect two arguments (or poles). In a similar approach, Yao *et al.* (2011) build graphs based on the relations discovered in text, considering that the trigger is the sequence of words located on the arcs making the path between the connected entities

3.1. Methodology

In this study we relied upon a set of principles for relation inventories presented in Vivi Năstase's book (Năstase, 2013). Among them, used as guidelines, there are: the set of relations should have a good coverage, provide useful semantic information, and entity classes should be well defined, with no overlapping.

The steps of our endeavour have been: pre-processing the Corpus; annotating entities; annotating semantic relations; evaluation. Here, we will present the annotation conventions for all the semantic relations types and subtypes, and finally, a critical evaluation of our Corpus.

3.1.1 Corpus pre-processing

As text for our corpus we used a Geography manual (Neguț *et al.*, 2008), containing almost 31,000 lexical tokens. In our vision, this is meant to be a first version of a larger Romanian corpus dedicated to geographical entities and relationships.

On the initial PDF format, which included images and various formatting to improve its attractiveness for pupils (being an 8th grade textbook) we applied boilerplate techniques in order to obtain the raw unformatted text. These included the extraction of text from pdf, removing formatting information and additional inserts (page number, image captions, others), correcting diacritics and a final manual correction of some errors in the resulted text.

The annotation process is a long and complex one, manual annotation being preceded by the following automatic annotation steps: POS tagging (Simionescu, 2011), NP-chuncking (Simionescu, 2012), NER (Name Entity Recognition) (Gîfu and Vasilache, 2014). The chain of these modules facilitates the work of human annotators, by making easy the identification of entities (supposed to be members in relations).

3.1.2 Annotating the corpus

The resulted document is then manually annotated for geographical entities, and then geographical relations, using a purposely-built annotation tool (see Figure 1). A working session with our annotation tool, RelAnn (*Relation Annotation*) begins by loading the file (an XML format), which triggers the graphical highlighting of the automatically annotated entities (each type in a different colour). During the session the human annotator is allowed to correct annotation of entities and to add/correct annotation of semantic relations.

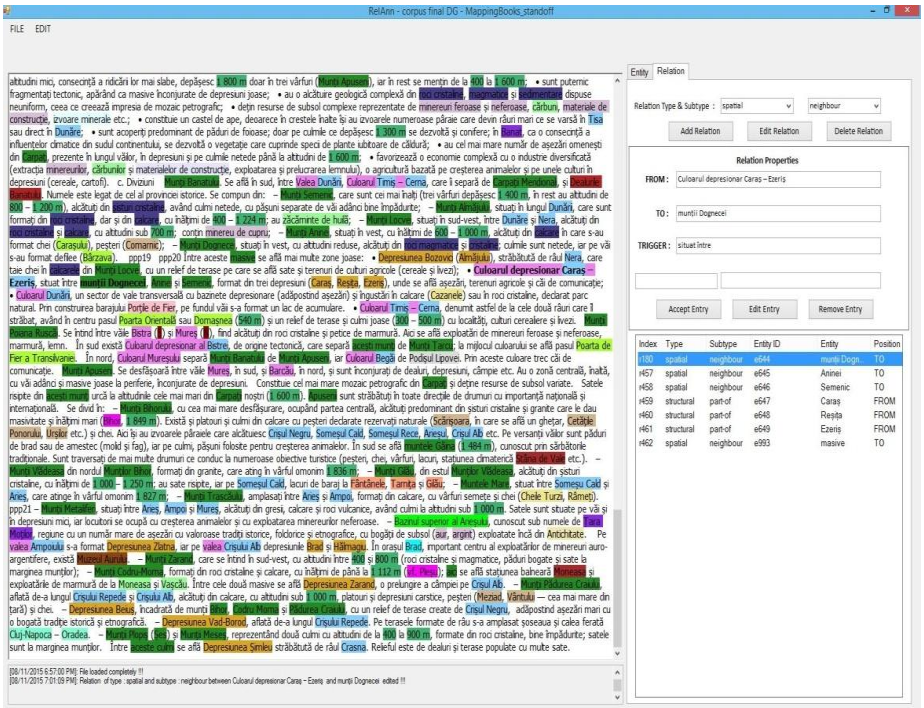


Figure 1. A snapshot of a working session with RelAnn

3.2 The inventory of semantic relations

In the MappingBooks project we focussed on 15 entity types, subcategorised in 105 subtypes[4], and three types of semantic relations (referential, spatial and structural), 17 subtypes. They will be described below. Each relation holds between two *arguments* (also called *poles*), and one word or expression, which signals the type of relation, called *trigger*.

[4] Person; Location (Village, Town, City, County, Region, Resort, Neighborhood, Street, Island, Continent, Archipelago, Peninsula, Cape, Coast, Oasis, Forest, Autoroute, Port, Airport, Train_station, Customs, Protected, Country, Empire); Geo_Position (Cardinal, Parallel, Meridian, Ecuator, Emisphere, Poll); Geology (Rock, Era, Strata, Basin, Platform, Orogene); Landform (Physiographic_ region, Plain, Hill, Mountain, Peak, Plateau, Floodplain, Depression, Aisle, Pass, Cave, Glacial_circus, Volcano, Mud_volcano, Levees, Terrace, Steppe, Beach, Valley); Clime (Temperature, Precipitations, Wind, Region); Water (Lake, Brook, River, Channel, Sea, Basin, Branch, Delta, Ocean, Spring, Strait); Dimension (Latitude, Longitude, Height, Depth, Length, Surface, Currency, Quantity, Percent); Organization (Education, Party, ONG, Gov, Army); URL; Timex; Resource (Coal_basin, Wyne, Fruits, Minerals, Metals, Mineral_water); Industry (Thermopower, Hydropower, Atomicpower, Oil_refinery); Cultural (Monastery, Church, Castle, Museum, Park, Monument); Unknown.

Our notations are expressed in XML stand-off. We use <S> for marking sentences, with the attributes: *id* and *start/stop offsets*, and <W> for words with attributes like *id*, *lemma*, *morphosyntactic features*, the *start/stop offsets* and the *text* itself. <ENTITY> marks entities with attributes as: *id*, *type*, *subtype*, *reference to word*, *colour* and *start/stop offset*s. Finally, the manual annotation adds to the file <RELATION> markers, with the attributes: *id*, *type*, *subtype*, *from*, *to* and *trigger* (excepting the *coref* types, which do not include triggers). Bellow, each type of relation will be exemplified.

I. ***Referential relations*** with four subtypes are listed and illustrated below:

- *coref* – indicates anaphoric coreferences.

```
… sudul 1:[Europei] … nordul 2:[acesteia] (En: … south of 1:[Europe]… north of 2:[it]) => <RELATION TYPE="REFERENTIAL" SUBTYPE="COREF" FROM="2" TO="1"/>;
```

The corresponding XML – stand-off format:

```
    <W ID="w18.13" LEMMA="Europei" MSD="Np" POS="NOUN" Type="proper" offsetStart="3512" offsetStop="3519" text="Europei" />

  <W Case="oblique" Gender="feminine" ID="w18.23" LEMMA="acesta" MSD="Pd3fso" Number="singular" POS="PRONOUN" Person="third" Type="demonstrative" offsetStart="3562" offsetStop="3570" text="acesteia" />

    <ENTITY ID="1" SUBTYPE="continent" TYPE="location" WORDSID="w18.23" Color="Chocolate" offsetStart="3562" offsetStop="3570" />

  <ENTITY ID="2" SUBTYPE="continent" TYPE="location" WORDSID="w18.13" Color="Chocolate" offsetStart="3512" offsetStop="3519" />

  <RELATION ID="r22" SUBTYPE="coref" TYPE="referential" TRIGGER="" FROM="1" TO="2"/>
```

- *isa* – links elements to their classes (concepts).

```
Așezarea geografică a 1:[României]… 3:<este> o 2:[țară dunăreană] => <RELATION TYPE="REFERENTIAL" SUBTYPE="ISA" TRIGGER="3" FROM="1" TO="2"/>;
```

- *feature-of* – describes a characteristic of an entity.

```
...cu 1:[stepa], 3:<caracteristică> 2:[Europei Estice] => <RELATION TYPE="REFERENTIAL" SUBTYPE="FEATURE-OF" TRIGGER="3" FROM="2" TO="1"/>;
```

II. ***Spatial relations*** with six subtypes are listed and illustrated below:

- *distance* – marks the distance between two entities.

```
Aceasta înseamnă că 1:[România] se află... - 3:<la jumătatea distanței> între 2:[punctele extreme ale Europei]=> <RELATION TYPE="SPATIAL" SUBTYPE="DISTANCE" TRIGGER="3" FROM="2" TO="1"/>;
```

- *near* – evidences the explicitly expressed closeness between two entities.

```
...1:[țara noastră] este mai 3:<aproape de> 2:[Marea Mediterană]... => <RELATION TYPE="SPATIAL" SUBTYPE="NEAR" TRIGGER="3" FROM="1" TO="2"/>;
```

- *far* – evidences the explicitly expressed farness between two entities.

```
1:[țara noastră] 3:<decât de> 2:[Oceanul Arctic] => <RELATION TYPE="SPATIAL" SUBTYPE="FAR" TRIGGER="3" FROM="1" TO="2"/>;
```

- *position* – indicates the relative position between two entities involving the mention of a cardinal point.

```
1:[Grupa Nordică].... Este limitată la nord de granița cu Ucraina, iar 3:<la sud> de 4:[Depresiunea Dornelor], 2:[pasul Mestecăniș] => <RELATION TYPE="SPATIAL" SUBTYPE="POSITION" TRIGGER="3" FROM="1" TO="4 2"/>
```

- *neighbour* – indicates that two geographical zones have a common border.

```
1:[CARPAȚII ORIENTALI] - se desfășoară între 3:<granița> cu 2:[Ucraina]... => <RELATION TYPE="SPATIAL" SUBTYPE="NEIGHBOUR" TRIGGER="3" FROM="2" TO="1"/>;
```

- *intersection* – linearly shaped entities that intersect.

```
… la 9:<intersecția> axei mărilor 2:[Marea Mediterană] – 3:[Marea Neagră] – 4:[Marea Caspică] cu axa fluviilor și canalelor 6:[Rhin] – 7:[Main] – 8:[Dunăre]... => <RELATION TYPE="SPATIAL” SUBTYPE="INTERSECTION" TRIGGER="9" FROM="2 3 4" TO="6 7 8"/>;
```

III. ***Structural relations*** with seven subtypes are listed and illustrated below:

- *part-of* – indicates that one entity is a constituent part of another.

```
Pe 2:[continentul european], 1:[România] este 3:<situată>... => <RELATION TYPE="STRUCTURAL" SUBTYPE="PART-OF" TRIGGER="3" FROM="1" TO="2"/>;
```

- *confluent-of* – expresses the branch for rivers.

```
1:[reţeau de râuri]... 3:<ajungând la> 2:[Dunăre] => <RELATION TYPE="STRUCTURAL" SUBTYPE="CONFLUENT-OF" TRIGGER="3 " FROM="1" TO="2"/>;
```

- *source* – mentions the source of rivers.

```
din 1:[Podişul Huedin] 3:<izvorăşte> 2:[Crişul Repede]. => <RELATION TYPE="STRUCTURAL" SUBTYPE="SOURCE" TRIGGER="3" FROM="2" TO="1"/>;
```

- *has-surface* – specifies that an entity has a certain surface.

```
1:<Suprafaţa> 2:[ţării noastre] este de 3:[238.391 de km2]. ⇒ <RELATION TYPE=”STRUCTURAL” SUBTYPE=”HAS-SURFACE” TRIGGER="1" FROM=”2” TO=”3”/>
```

- *has-lenght* – shows the length of a particular linearly shaped entity.

```
1:[Carpaţii] constituie unul din lanţurile montane europene însemnate, cu o 3:<lungime> de peste 2:[1 500 km] => <RELATION TYPE="STRUCTURAL" SUBTYPE="HAS-LENGTH" TRIGGER="3" FROM="1" TO="2"/>;
```

- *has-height* – sets the height of a mountain, etc.

```
… crestele] 1:[munţilor Bucegi] şi 2:[Pietrei Craiului] 3:<depăşesc> 4:[2 000 m] => <RELATION TYPE="STRUCTURAL" SUBTYPE="HAS-HEIGHT" TRIGGER="3" FROM="1 2" TO="4" />.
```

- *has-value* – expresses a numerical value different than height, length, surface.

```
Populaţia 1:[României] 3:<numără> 2:[8.6 milioane de locuitori]… => <RELATION TYPE="REFERENTIAL" SUBTYPE="HAS-VALUE" TRIGGER="3" FROM="1" TO="2"/>;
```

4. Statistics and interpretations

From the entire set of semantic relations presented above, our Corpus highlights values for 15 entity types, and 3 semantic relations types. The annotation itseld was performed by two researchers and one student, and lasted not less than a year. The hierarchy of types and subtypes was enriched and refined several times during this period and represents a complete set only with regard with the application at hand – the MappingBooks project and the tekst we used as a hub document in the project. We are aware that it is very probable other texts will require more types/subtypes.

Table 1 presents a statistics of the entities annotated in the whole corpus (which includes 30,437 lexical tokens). The data confirm that predominant entity types are entities largely mentioned in Geography manuals (`Location`, `Landform`, `Water`, `Dimension`).

Table 1. Statistics over the manually annotated geographical entities

Type	**No.**	**%**
Location	1641	37.43
Landform	1076	24.54
Water	587	13.39
Dimension	374	8.53
Resource	170	3.88
Timex	107	2.44
Clime	96	2.19
Geology	94	2.14
Cultural	75	1.71
Geo_Position	65	1.48
Person	64	1.46
Industry	35	0.80
Organization	0	0
URL	0	0
Unknown	0	0

The semantic relations annotated in our corpus are shown in Table 2. Actually, the situation is similar, being known that `spatial` and `structural` relations have a large impact in a Geography textbook.

Table 2. Statistics over the manually annotated semantic relations

Type	**No.**	**%**
Referential	964	46.04
Structural	731	34.91
Spatial	399	19.05

5. Conclusions and discussion

The work described in this paper is a step towards developing an automated process of annotating geographical entities in texts and semantic relations that link them. For now, the process involved only manual annotation supported by an automated pre-processing step.

The work reported here, of inventorying, classifying and annotating entities and relations ended up in the realisation of a resource that will have a crucial contribution as part of an on-going project (MappingBooks), aiming to offer to the user enhanced interaction with an e-book. Also, such a resource can serve as training data for a tool designed to automate an analogous annotation effort.

The manual annotation activity also puts in evidence other types of semantic relations than those presented in this paper, which will have to receive some attention in the future: expression of negated facts (e.g., *România ... nu face parte*

din Peninsula Balcanică/(En: *Romania ... is not part of the Balkan Peninsula*), temporal relations, a topic with a significant impact in NLP (e.g. *În Carpați… ghețari în urmă cu 10 000 – 300 000 ani*/(En: *In the Carpathians ... glaciers 10 000 – 300 000 years ago*), or the old toponyms that are no longer in use (e.g. the ancient city names, such as: Histria, Tomis, Callatis, Apulum, Ampelum, Napoca, Potaissa, Sucidava), etc.

In order to generalize and fully automatize the process described in this paper we will need to perform similar studies on other relevant texts, such as travel guides, which may lead to the inclusion of additional relation types. We are also aware that for the recognition task we will have to include additional layers of automatic annotation, among which resolution of anaphora and syntactic analysis may be of a substantial importance.

Acknowledgements

This work was done with the support of the PN-II-PT-PCCA-2013-4-1878 Partnership PCCA 2013 MappingBooks grant. The RelAnn tool was developed by Alexandru Sălăvăstru, a student in the Master of Computational Linguistics at the Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iași.

References

Blessing, A. and Schütze, H. (2010). Fine-Grained Geographical Relation Extraction from Wikipedia. In *Proceedings of LREC-2010*, pp. 2949-2952 Valletta, Malta.

DeLozier, G., Baldridge, J. and London, L. (2015). Gazetteer-independent toponym resolution using geographic word profiles. In *Proceedings of AAAI 2015*. The AAAI Press.

Gîfu, D. and Vasilache, G. (2014). A language independent named entity recognition system. In *Proceedings of ConsILR-2014*, M. Colhon, A. Iftene, V. Barbu Mititelu, D. Cristea, and D. Tufiş (eds.), "Alexandru Ioan Cuza" University Publishing House, Iaşi, pp. 181-188.

Egenhofer, M., and Herring, J. (1993). Quering a Geographical Information System. In *Human Factors in Geographical Information Systems*, D. Medyckyj-Scott and H. Heanshaw, Belhaven Press, London, pp. 124-136.

Egenhofer, M., and Mark, D. (1995). Naïve geography. In *Spatial Information Theory. A theoretical basis for GIS*, Frank, A.U., Kuhn, W. (eds.), Int. Conference COSIT'95, LNCS 988, Springer, Berlin, pp. 1-15

Klien, E. and Lutz, M. (2005). The Role of Spatial Relations in Automating the Semantic Annotation of Geodata. In: Cohn, A. and Mark, D (eds.). *Proceedings of the Conference of Spatial Information Theory (COSIT 05)*, Ellicottville, NY, USA. Lecturer Notes in Computer Science, Vol. 3693, pp. 133-148.

Leidner, J. L. (2007). *Toponym resolution in text: Annotation, evaluation and applications of spatial grounding*. SIGIR Forum, 41(2), pp. 124–126.

Lieberman, M. D., Samet, H., and Sankaranarayanan, J. (2010). Geotagging with local lexicons to build indexes for textually-specified spatial data. In Feifei Li, Mirella M. Moro, Shahram Ghandeharizadeh, Jayant R. Haritsa, Gerhard Weikum Michael J. Carey, Fabio Casati, Edward Y. Chang, Ioana Manolescu, Sharad Mehrotra, Umeshwar Dayal, and Vassilis J. Tsotras (eds.), *ICDE*, IEEE, pp. 201–212.

Mani, I., Harris, D., Hitzeman, J., Quimby, R., Richer, J., Weller, B. (2008). SpatialML: Annotation Scheme, Resources and Evaluation. In *Proceedings of LREC-2008*.

Mani, I., Doran, C., Harris, D., Hitzeman, J., Quimby, R., Richer, J., Wellner, B., Mardis, S., and Clancy, S. (2010). SpatialML: annotation scheme, resources, and evaluation. In *Language Resources and Evaluation*, 44(3), pp. 263–280.

Năstase, V., Nakov, P., ´O. S´eaghdha, D., Szpakowicz, S. (2013). Semantic Relations Between Nominals. Morgan & Claypool Publishers, California (USA).

Neguț, S., Apostol, G., Ielenicz, M. (2008). Geografie, Humanitas Educațional, București.

Pereira, G. M. (2002). A Typology of Spatial and Temporal Scale Relations, vol. 34 (1), pp. 21-33.

Roberts, K. and Harabagiu, S.M. (2012). UTDSpRL: A joint approach to spatial role labeling. In *Proceedings of SemEval 2012*, pp. 419–424.

Roberts, K., Skinner, M., and Harabagiu, S. M. (2013). Recognizing Spatial Containment Relations between Event Mentions. In *Proceedings of the International Conference on Computational Semantics (IWCS)*, Potsdam, Germany.

Shariff A. R., Egenhofer M., Mark D. (1998). Natural-Language Spatial Relations between Linear and Areal Objects: The Topology and Metric of English-Language Terms. In *International Journal of Geographical Information Science*, vol. 12, pp. 215-246.

Simionescu, R. (20110. *UAIC Romanian Part of Speech Tagger*, resource on nlptools.info.uaic.ro, "Alexandru Ioan Cuza" University of Iași.

Simionescu, R. (2012). Romanian deep noun phrase chunking using graphical grammar studio. In: Moruz, M. A., Cristea, D., Tufiș, D., Iftene, A., Teodorescu, H. N. (eds.) *Proceedings of the 8th International Conference Linguistic Resources And Tools For Processing Of The Romanian Language*, pp. 135–143.

Speriosu, M. and Baldridge, J. (2013). Textdriven toponym resolution using indirect supervision. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, Vol. 1, pp. 1466–1476. Association for Computational Linguistics.

Yao, L., Haghighi, A., Reidel, S., McCallum, A. (2011). Structured Relation Discovery using Generative Models, In *Proceedings of EMNLP*.

Zhang, W. and Gelernter, J. (2014). Geocoding location expressions in twitter messages: A preference learning method. In *Jornal of Spatial Information Science*, 9(1), pp. 37-70.

PML: A PUNCTUATION SYMBOLISM FOR SEMANTIC MARKUP

IOACHIM DRUGUS

Institute of Mathematics and Computer Science, Academy of Sciences of Moldova

ioachim.drugus@math.md

Abstract

The first markup languages appeared as annotation symbolisms to describe presentation of data, but with advent of ontologies and Semantic Web, the need for "semantic markup" emerged, and a large family of XML-based standardized languages were designed for this purpose. In this paper, the focus is shifted from annotation by use of XML tags, which add metadata, towards disambiguation of natural language texts by use of their *native* markup devices, the punctuation marks, which do *not* add metadata. In other papers, the author proposed a symbolism "A3" as a compositional counterpart of Tim-Berners Lee's "Notation3" (N3) for authoring ontologies. This symbolism is extended here to be applied to natural languages as "Punctuation Markup Language". PML is ultimately intended for disambiguation of arbitrary natural language texts, but at this stage of development, it helps the disambiguation of only expressions shorter than simple sentences, i.e. it can be used "intra-propositionally". With current version of PML, the role to disambiguate the texts with at least one predicate is left to propositional and predicate logics, but a goal of future research is to extend PML for disambiguation of arbitrary texts. This paper is a presentation of a work in progress intended as first "request for comments" (RFC) on general features of PML – comments that could help complete the work on PML version 1.0.

Keywords: annotation, count-noun, markup, mass-noun, metalingua, PML, ontology, semantic markup, Semantic Web, sense, reference.

1. Introduction

The symbolism "Notation3" ("N3"), was designed by Tim-Berners Lee, the founder of WWW and Semantic Web, for authoring ontologies (Tim-Berners *et al.*, 2000) by humans in a standardized manner, so that these can be processed by the programs called "reasoners". The semantics of N3 is wider than the semantics of other standardized symbolisms (called also "languages"), such as RDF, RDFS, or OWL, but due to limited inventory of expressive means, N3 was not recommended by W3 Consortium as a standard for authoring Semantic Web ontologies. Nevertheless, the N3 symbolism remains as the main full-fledged symbolism for research of semantics of web-related ontologies. The main inconvenience of N3 in its application to

natural language is that it does not comply with compositionality principle stating that the meaning of an expression is a function of meanings of its constituents.

A set-theoretic approach to ontologies of Semantic Web shaped up in papers (Drugus, 2007, 2009a, 2009b, 2010, 2012), which uses a symbolism "A3", pivoted around three set-theoretic operations – "atomification", "aggregation" and "association", used to form the three main types of entities which occur in the universe of set theory – atoms, sets and ordered pairs. The A3 symbolism is compliant with compositionality principle in the same manner as natural languages are, and it was proposed as a "compositional" counterpart of N3. Subsequently A3 was enhanced to a symbolism to fit for natural languages specifics, and the new symbolism is presented here as "Punctuation Markup Language", or "PML" – a language for "punctuation markup".

The PML markup can be opposed to "tag markup" of XML used in standardized languages of Semantic Web by one feature – PML does *not* add metadata to the text, whereas XML was specifically designed to add metadata. In markup process, the symbols of PML replace the native punctuation marks or the expressions which play the role of punctuation, and add PML "meta-punctuation" when the "native punctuation" is missing. Hence, PML can be considered as a "proof-reading" markup language. A sample of PML markup follows next, where a Romanian sentence is marked up in two manners to illustrate two various graphic presentations, called here "renditions", and its English translation in one such rendition illustrating only the PML markup:

„La scăldat!" zise Ion Chistruia, dar echipa lui, Gicu și Mihai, deja era pe ducă.
([La·scăldat!]**·:(**zise**·:(**Ion**·::**Chistruia)))**,·**(((<echipa>**·:**lui)**:**·(<Gicu**,**·Mihai>))**·:(**deja**:**·(era**·:**<pe·ducă>)))
([La scăldat!] **:(**zise **:(**Ion **::**Chistruia)))**, (((<echipa> **:**lui)**:** (<Gicu**,** Mihai>)) **:(**deja**:** (era **:**<pe ducă>)))
„Swimming!" called Ion Freckle, but his team, Gicu and Mihai, was up to it already
.
([Swimming!] :((called: (Ion **::**Freckle))), (((his: <team>): <Gicu, Mihai>) :(((was :<up to>): it) :already))

The first kind of rendition uses throughout the text the character "·" ("interpunct") instead of the character "space" in the second rendition, and this is why this rendition looks "overloaded" with punctuation marks, whereas the second rendition looks closer to the original text, except of the colon (which is partially used as in N3; actually this is a "generalized use"). If a text is written in MS Word, then to toggle between the two renditions, one presses the button "paragraph". Notice, that in informatics, the term "punctuation mark" has a wider extension (https://en.wikipedia.org/wiki/Punctuation) and, in particular, the characters "space", "interpunct" and "parentheses" are termed "punctuation marks". In this paper, this term refers to punctuation marks used outside words (thus, the apostrophe or the dash is not referenced here as "punctuation marks").

One can easily guess the functions of symbols in “the sample”. Thus, (a) parentheses serve for disambiguation; (b) the colon marks up the “modifier” – if the “modifier” is before the “modified”, then the colon is post-fixed, and if the “modifier” is after the “modified”, the colon is pre-fixed; (c) the punctuation mark “**::**” (“double colon”) marks up an “alias” (“poreclă”), (d) the angular brackets indicate that the enclosed expression should be treated as idiomatic, (e) the square brackets play the role of quotes used for direct speech or reference to a text. The Romanian sentence presented here will be further referenced as just “the sample”. The final remark about this “sample” relates to rendition – whether or not to use an interpunct instead of space within a text enclosed between quotes is a matter of taste and this has nothing to do with PML markup.

Real-life disambiguation of texts presupposes understanding them – an aptitude of the intellect. Thus, it would take application of methods of artificial intelligence to properly model this process in software. Nevertheless, a program built by my students for disambiguation of English texts by use of A3 symbolism simulated an “understanding” above statistical expectations, and there is a chance that PML with its superset of the set of A3 symbols will display a higher “intelligence”

The symbols of PML are treated as meta-symbols of punctuation marks native to natural languages and a device for uniform mapping of native punctuation marks to such meta-symbols would be handy. This device is provided by the practice to use a space after each punctuation mark. Thus, with “interpunct rendition”, where the interpunct stands for space, the compound punctuation mark “**,·**” can be treated as a symbol of PML, a meta-symbol for comma. Thus, the function to symbolize the mapping of native punctuation marks to meta-symbols is assigned here to the interpunct**.**

2. A view upon composition, markup and disambiguation

A text is said here to be “composed” by a “writer” and “read” by a “reader”. The writer composes the text using “composition operations”, and this result in a denotation of a superposition of these operations, the final text. The reader restores the order of application of composition operations in a process called “disambiguation”, and then he interprets the text to finally obtain a “reading”. The text might not provide enough data to restore precisely the “reading” intended by the writer. Therefore, the reader often faces the choice of one reading from many possible ones. The writer also encodes via punctuation some instructions, and the reader executes them to obtain one “reading” or another of the text.

The meaning is “constructed” by the writer, and is “reconstructed” (or “construed”) by the reader, but it resides in their minds (or “memories” of intelligent machines); what is exchanged between them is the text which is ink on paper or electrical signals. Therefore, one can treat the reader and the writer as text processors, intelligent enough to process punctuation marks only. One can consider them implemented in human mind and called as „subroutines” when writing or reading.

The punctuation marks are the only means of markup available to a writer of a natural language text, but the reader mentally adds additional markup to disambiguate the text. Assuming that this mental markup is reflected by PML symbols, consider several examples of instructions to the reader encrypted in „the sample". The square brackets in the text „[Hai·la·scăldat!]" denotes the following instruction to the reader „Do not construe the enclosed text – treat it as plain text". The angular brackets in the text „pe·ducă", on the contrary, instructs the reader: „Construe the enclosed text as a whole – do not reduce its meaning to the meanings of its constituents". Both these instructions imply that certain defaults are overridden. Namely, the square brackets demand overriding the default established in all natural languages – the default to interpret the text, and the angular brackets demand to override compositionality – another default for natural languages. The reason why in this paragraph the word „construe" instead of „interpret" was used, is that „construe" is treated as a wider process: to construe is first to disambiguate and then to interpret.

There are defaults established specifically by linear scripts, like those of European languages. These scripts impose the default to consider two adjacent words as a set-theoretic ordered pair - a default, which is usually overridden by the comma. Another default established by natural languages is on how to relate „modification" with the order.

To explain this, in Romanian this kind of default is to place the modifier after the modified, and in English, this default is to place the modifier before the modified. In PML, the symbol „:" is used to override the default established by any language regarding such „modification" (of meaning and reference). Finally, the symbol „·::" denotes a complex instruction in general case, but in the expression „Ion·::Freckle" it simply instructs to consider „Freckle" same thing as „Ion".

The expression „Ion Frecke" could be incorrectly interpreted „by default" as „Ion·:Freckle", where „Freckle" is a modifier. But the symbol „·::" overrides this default and instructs the reader to treat „Freckle" as another name of Ion „by definition". Here, the semantics of PML symbols was specified in terms of operations done by the „reader" according the punctuation instructions, and this can be called „operational semantics". There is also an „extensional semantics" of PML which will be explained in 3.4. („extensional" is a synonym for „set theoretic").

3. Metalingua

PML is used for disambiguation of texts and this is treated here as restoring the order of application of composition operations done by the writer. Therefore, to fully understand the mechanism of disambiguation and be able subsequently to materialize this mechanism in software, one would need to study the algebra of these operations and the logic of this algebra (the „logic" of algebra is the set of formulas valid in the algebra). The language of these operations (or the „signature" as the algebraists would name this) is called here „metalingua".

The denotation for metalingua used here is „mL", since the acronym „ML" cannot be used due to its extensive use in linguistics for other purposes. Also notice that „mL" can be used only inside a sentence; it does not look good in the beginning of a sentence, and MS Word tends to automatically change it to „ML" – in the beginning of a sentence one has to write „Metalingua". A difference between mL and PML is that mL has a vocabulary, whereas PML has only symbols of operations used for markup.

The composition operations are applied to texts, their application results in texts, and this is one „level" of their action – the „syntactic level". But these operations are also used to encrypt a meaning and this is another level of their action – the „semantic level". The semantic level will be in our focus next. A text can be treated as a „vehicle", a „container", and its „content" – as consisting of „subject matters" of two kinds, meaning and information (mL ignores other kinds of content, like that called „fascination").

The information is considered here as subject matter of a text which contains at least one predicate. The minimal vehicle of information is a „clause" of a sentence or a „simple sentence" (a sentence containing exactly one clause). Thus, information is treated here as subject matter of what logicians call „propositions" (the abstract images of sentences), and this subject matter is treated in logic by propositional logic and predicate logic. On the other hand, the meaning in its „pure form" is „intra-propositional". A vehicle of meaning is named here „expression" and an expression is said to „express" a meaning; the mL version 1.0 is a language of expressions. One can expect that mL will be developed to the next version enabled to treat also the texts *not* shorter than a clause – namely, a clause, a complex sentence, or a many-sentence texts. A „generic" term for these terms is „predications". The predications are made up of expressions.

Notice, that „the sample" is a complex sentence (a predication), with two clauses (predications), and the compound symbol „,·" separates them. The semantics of this symbol is quite different from semantics of this symbol's occurrence in expression „Gicu,·Mihai". The first occurrence of this symbol is said here to be within a „predication context" and the second – within an „expression context". Also, notice, that the symbols of the last applied operations which form the two clauses are symbols of predicates, and their semantics differs from the semantics of same symbols used elsewhere in „the sample".

The semantics of each PML symbol depends on the context in which it occurs, which can be either „expression context" or „predication context". PML version 1 is about the expression context. The predication context was mentioned here to delimit the focus of PML version 1.0 „from outside", but also to explain the use of all PML symbols which occur in „the sample". The mL has four components: (1) a universal vocabulary, (2) a composition framework, (3) a semantic framework, (4) an extension framework. The framework for extension of mL is rather complex and the topic (4) will not be discussed here.

3.1 The universal vocabulary

The term „*vocabula* of a language" is used here to reference a text, whose structure is not analysed by the writer but is used as a „building block" in the process of composing other texts. One must define the vocabulary of mL in such a manner that arbitrary texts can serve as vocabulas, but then, the writer must have a means to indicate to the reader that the mL's *characters* (out of which are built mL's symbols) should be „escaped" so that they do not „function". One of the „escaping procedures" is to use an „escape character", like backslash „\", in front of such an mL character. Having established a procedure of escaping, the vocabulary of mL is considered to be the (infinite) set of all strings of Unicode characters with all occurrences of mL characters escaped.

A vocabula of mL can contain spaces separating the „continuous" strings, the „words", and one may wonder why „vocabulas" were not defined as just such „words", so that the texts could be considered as built of them by mL operations. One of the reasons for this is that the texts serving as vocabulas are intended to serve as objects of the metalinguistic analysis done by use of mL, and it is convenient that these are vocabulas of language. This treatment forces to treat mL vocabulas as „unanalysed texts", whereas the expressions built of them as „analysed texts". Due to the fact that the texts in any natural language can be encrypted in Unicode, this is called „universal vocabulary".

3.2 The composition framework of mL

A composition operation of mL can be a superposition of others, and then it will be said to be "reducible" to them. A set *B* of mL operations is said to be a *basis* of mL, if all the mL operations are reducible to some of operations in *B*. If no operation in a basis *B* is reducible to other operations in *B*, then *B* is said to be an "orthogonal basis" of mL. An orthogonal basis of mL can be selected in various manners. Here, the operations in the basis are selected based on reasons of extensional semantics, i.e. based on their interpretations as set-theoretic operations. Also, the composition operation are named mainly according its role in set theory.

The symbols of PML are symbols of mL operations and these operations are the three unary operations ("association", "atomification", "individuation") for which the round, square, and angular brackets, are used, respectively; a unary operation ("definiendum") with "**::**" markup to indicate that the operand is a "definiendum" ("the defined"); a binary operation ("aggregation") with infix notation "**,·**", and two symmetric operations of "modification" with infix notations "**·:**" and "**:·**", respectively. The use of these operations will be explained below in a more systemic manner than before.

To distinguish between atomification and individuation one needs to take into account the distinction between "use" and "mention" (Quine, 2003, p. 23). Namely, an expression can be "used", i.e. used to reference something *different* from itself, and an expression can be "mentioned", i.e. it can be referenced by another expression. Furthermore, there are two ways to mention an expression:

(a) it can be mentioned *ad litteram* via direct speech using quotes, and in this case (together with quotes) it references *itself*, or

(b) its meaning can be delivered by indirect speech.

The complex phenomenon of indirect speech is not treated by Semantic Web and this topic is not treated here. Thus, when the term "mention" is used here only in "ad litteram" sense. An expression which refers to a text is called here "literal" (same as in RDF). The quotes used in language display only one manifestation of the operation called „atomification". The semantics of atomification is more complex that the semantic of quotes and is termed „quasi-quotation" by Quine (1981) who developed it according his approach – here it was independently developed according another approach, which looks wider. The quasi-quotation or atomification operation is mechanism for „uniform" handling of the mention-use distinction (see also https://en.wikipedia.org/wiki/Quasi-quotation).

The *atomification* and *individuation* operations are *unary* operations – operations, which assign meaning to their operands. These are said here to always modify the meaning because of the specifics of our approach. Thus, "formal expression" is said here to carry a "formal meaning" or to be used in the "formal sense" and, thus, to also have a meaning – the "formal sense". This treatment allows considering all the constituents of a meaningful expression to be meaningful, including the constituents enclosed between quotes, and this permits to apply compositionality principle to all natural language expressions. The expressions enclosed between quotes are said to be "mentioned" in contrast with other expressions, which are said here to be "used".

To refer to a phrase, the linguists enclose it between square brackets and indicate its grammatical category as a subscript like this: [the person, who smiles]$_{NP}$ (here the subscript "NP" is an acronym of "Noun Phrase"). This notation reflects a complex metalinguistic device, but a part of it – the square brackets, is considered as a compound symbol of a mL, a symbol of a composition operation called here "atomification".

There are many kinds of quotes in natural languages and all them latently use atomification. So, the word "team" used in "the sample" is used for irony, and one can say that the reader of this text has to re-interpret this word, i.e. "erase" its original meaning (i.e. apply atomification) and assign another meaning – here, that of a "would-be team".

The natural languages do not mark up the expressions *used* in a text (*not* "mentioned"), but this is also considered here as a composition operation; it is marked up with angular brackets and is named "individuation". An expression enclosed between angular brackets is supposed to be interpreted as an "individual". The words in a text refer to individual objects, "individuals", and due to default established by languages, they do not need be marked up with angular brackets. But when a word is intended to have a modified meaning (like "team" in the sample), one can use angular brackets to specifically indicate this. But the many-words expressions intended to carry a meaning "as a whole" (i.e. to be "idiomatic

expressions") need to be enclosed between angular brackets to indicate this fact to the reader.

In natural languages, the meaning of expressions is strongly context-sensitive and the minimal context of a word is the adjacent word. By default, two adjacent words represent a context modifying both the meaning and the extension of each other (if only this default is not overridden by a comma in between). Therefore, the space between words is treated here as denoting an operation called here "modification operation".

In early Latin, before the space started being used for separation of words, a punctuation mark like this "·", called "interpunct" served this role, and a vestige of this is the MS Word UI, where the interpunct, alongside other "invisible" symbols, appear in a text each time when the button "paragraph" is pressed. Consider two expressions with same meaning – the English expression "white house" and its Romanian equivalent "casă albă". These expressions will be treated here as obtained in result of application of the modification operation – an application represented differently, i.e. it has two "presentations" ("direct presentation" like in Romanian, and "inverse presentation" like in English. Its operands are named "modifier" and "modified".

The compound symbol "**·:**" is used in *direct* presentation and the symmetric symbol "**:·**" in the *inverse* presentation, of modification, where interpunct is on the side of the modified, and the colon is on the side of modifier. This complies with extensional semantics where the modified is interpreted as *one* individual (one dot), and the modifier as a class of *many* individuals (two dots).

The *association* operation is marked up by round brackets and is treated here as acting on the level of syntax. It is treated this manner only because the discourse here is about natural languages, but actually, association operation acts on the level of structure, which precedes syntax (meaning is assigned to structures by the atomification and individuation operations as this is stated above). Its name was derived from the practice to refer as "left associated" or "right associated" to mathematical expressions with brackets used for disambiguation of order of operations' application.

A formal language prescribing consistent use of brackets cannot produce ambiguous texts. Also, only one kind of brackets can be used for disambiguation and in mL these are the round brackets, "parentheses". But mL does not prescribe to use them consistently, since it accepts also ambiguous texts. The result of markup is said here to be a "reading" and such a "reading" is not supposed to be a completely disambiguated text.

The parentheses are used in mL to instruct the reader to treat a part of text as "a whole" (for a purpose to be further indicated). The necessity to indicate this appears only "sporadically", since in many cases, a part of text is treated as "a whole" due to the "connectives" (the logical term for a symbol of an operation) used between words, which are other mL symbols. In mL, the association operation is treated as a

unary operation per the same mechanism as one defines the notion of 1-tupple in terms of the ordered pair.

3.3 The semantic framework

The semantics of the standardized languages of Semantic Web is rigorously defined in terms of set theory – a semantics which can be said to be „extensional semantics". The existing extensional semantics basically use the ZF set theory, a theory which discusses only about sets, a „pure set theory". The natural languages are more complex than the ontologies of Semantic Web and, in order that an extensional semantics is developed for natural languages, the „whole ontology" of objects which occur in various set theories is needed.

This ontology has the following main types of objects („object" is the term for the entities populating the universe of discourse of set theory): *set*, *atom*, *ordered pair* and *class*. Also, an *algebraic* set theory is needed for an extensional semantics, since the natural language texts are composed by (composition) *operations*. Currently, there exists no axiomatic set theory, which would be both algebraic and would contain all types of set theory ontology in its universe of discourse. This implies that the development of an extensional semantics of natural languages can go hand in hand with the development of a set theory appropriate for natural languages, and the intuitive contributions of linguists to this project can be same valuable as the contributions of logicians.

A view upon such a „semantic set theory" was presented by Rogers (2012), and some features of such a theory, in a more formal presentation, were described in (Drugus, 2015).

Expressions are built of „parts of speech" and a common term used here for any part of speech is „name"; a many-word expression is treated here as a „compound name". According to Frege (1892), but in terms convenient for our approach, a name „makes sense", or it „has a meaning", and is used to „refer" to one of its denotata (the term „referent" can be confusing). The meaning of a name N limits the multitude of its denotata, which in most cases, should be treated as a „class" rather than a set.

This multitude is denoted here as $\varepsilon(N)$ and is termed „extension of N". In any case, to specify semantics of names, one has to introduce a limitation of extensions of all names to a set, named „universal set". Therefore, one customarily specifies the semantics with *sets* as extensions of names rather than with classes. Two names can have different meanings, but same extension, and it is incorrect to consider „meaning" and „extension" as the same thing.

It becomes clear from this, that in a set theory appropriate to serve as a semantic framework for natural languages, the „extensionality axiom" should *not* be postulated, so that two different sets x and y can have same elements, i.e. $\varepsilon(x) = \varepsilon(y)$. Therefore, the names should also have another attribute, different from „extension", in order that they are distinguished. This attribute is called „intension" in (Nikitchenco *et al.*, 2012). In current approach, what is called „intension" is

expressed by the meaning of a name, and the names serve as „labels" of „extensions".

An interpretation of modification operation is this: $\varepsilon(x\cdot\!:\!y) = \{\varepsilon(x)\} \cup \varepsilon(y)$, and the "characteristic property" of this treatment is this: $x\cdot\!:\!y = y$ iff $\varepsilon(x) \in \varepsilon(y)$. Such a semantics complies with the treatment of names as vehicles of *meaning* (forgetting about its referential function), and the operation treated this manner called here "meaning modification" operation. Thus, according this treatment, "alb" is interpreted as the set of all white things, "casa" is interpreted as an individual thing, and the expression "casa·:albă" as an expression of a fact which is true only when the individual "casa" is member of the set "alb".

Another semantics can be obtained by treating the names as references (and "forgetting" they also are vehicles of meaning) expressed by the following characteristic property: "$x\cdot\!:\!y = x$ iff $\varepsilon(x) \in \varepsilon(y)$". The composition operation treated like this is called here "reference modification" operation. To specify such a semantics, one would use the following definition: $\varepsilon(x\cdot\!:\!y) = \{\varepsilon(x)\} \cap \varepsilon(y)$. These are two different manners of interpretation of modification, which apply "modification" to "meaning", and to "reference".

Without the notion of atom, any extensional semantics would incompletely reflect the meaning of expressions because of the proper names, which cannot be interpreted as sets without inflicting upon the intuition of an „atom" as a „non-set". To introduce atoms (and classes) in our discourse, a short presentation of these set-theoretic concepts follows in the spirit of „semantic set theory", where the notion of „meaning" or „sense" is dominant.

All mainstream set theories are presented in terms of one binary relation called „membership" and the fact that two objects *x* and *y* are in this relation is denoted as „$x \in y$". Logicians are interested in the *truth* of this relation for various couples of objects, but we are interested in the *meaning* of expression „$x \in y$". First, one would obviously ask what a set is in these terms, and the immediate answer is that a set is an object *s*, such that both expressions „$x \in s$" and „$s \in y$" make sense.

Having established this, one can define an *atom* as an object *a*, such that the expression $x \in a$ does *not* make sense, whereas the expression „$a \in y$" makes sense. To say that an atom is an object which does not have elements is incorrect, since one cannot state anything about something which does not make sense. It makes sense to say only about the empty set (which is a „set"), that it is an object which does not have elements. Obviously, one can now define the notion of class this manner: „a *class* is an object *c*, such that the expression $c \in y$ does *not* make sense", but the expression „$x \in c$" makes sense. This is how the three types of objects (set, atom, class) in the ontology of set theory can be defined from the perspective of natural language, i.e. in a semantic set theory.

In mainstream set theory, the term „individual" stands for an atom or a set (not for a class), and then an operation resulting in an individual *can* be called „individuation". Here, the individuation operation applied to an object *x* in the ontology of set theory

is denoted as $<x>$ and is defined like this: $<x> = \{x\}$, if x is a set, and $<x> = x$, if x is an atom or a class. This definition sounds natural for atoms x which are „Quine atoms" (i.e. x is such that $\{x\} = x$). This definition sounds as a novelty only due to classes. Whether or not one accepts the Church's treatment of „set" (Church, 1974), where this last treatment (for classes) is justified, does not matter for semantics, because in order to define the semantics of a language one would need to limit the discourse to a universe, and in such a universe all the objects of a language would be interpreted as sets.

Therefore, the treatment of the individuation operation as an operation also defined on classes should not raise concerns regarding the consistency (non-contradiction) of set theory. Having defined the individuation operation, one can extend the semantics defined over ZF of modification operation to arbitrary objects in the universe of discourse of set theory by using the symbol „<...>" instead of the symbol „{...}". In particular, one can define the semantics of „meaning modification" operation as $\varepsilon(x{\cdot}{:}y) = <\varepsilon(x)> \cup \varepsilon(y)$ and the semantics of „reference modification" operation like this: $\varepsilon(x{\cdot}{:}y) = <\varepsilon(x)> \cap \varepsilon(y)$.

An important question is whether or not the set theory can be used for specification of semantics of „formal expressions". Notice, that in set theory an atom is treated as any object in the universe of discourse which is not a class, and a formal expression *is* a non-class. Moreover, the formal expressions are the remakable atoms which are called „Quine atoms" since they refer to themselves. Therefore, to define to semantics of formal expressions, one would have to immerse the language texts into the universe of set theory. The atomification operation is intended to materialize this immersion, and this operation can be regarded as idempotent (i.e. $[x] = x$) for literals, Quine atoms.

In set theory, the term "aggregate" refers to collections of any kind - sets, classes, or other entities imagined as "multitudes" and from this term are derived the term "aggregation". In mL, the aggregation is treated as an operation applicable both to atoms and sets. In denotation of a finite set, the symbols of its elements are separated by commas, and this served as a reason to use comma as the symbol for the operation of placing two names one beside the other and consider that the meaning of the resulting text does not change when this order is changed.

The semantics of comma is the union operation: $\varepsilon(x{,}{\cdot}y) = \varepsilon(x) \cup \varepsilon(y)$. Notice, that if x and y are Quine atoms, i.e. if $x = \{x\}$ and $y = \{y\}$, then $\varepsilon(x{,}{\cdot}y) = \{x, y\}$. This might explain why in natural language, the students sometimes encounter difficulties in choosing the right word – "and" or "or" – when discussing about the union of sets. Probably, the natural treatment of the comma between two words is one where its meaning is expressed by the conjunction "and" when the words are nouns referencing material objects ("atoms"). But according to this treatment, the semantics of "and" *naturally* extends to the objects in set theoretic ontology to the meaning of "or" interpreted as union and not as intersection.

4. Future research and development

One of the intended uses of PML is authoring/presenting ontologies of Semantic Web in natural languages rather than complex standardized languages of Semantic Web. This use could make the Semantic Web a "democratic" tool similar to Wiki. Figure 1 shows one manner of materializing this "democratization" by development of software for various conversions (notice that only one tool is needed for conversion between PML and N3).

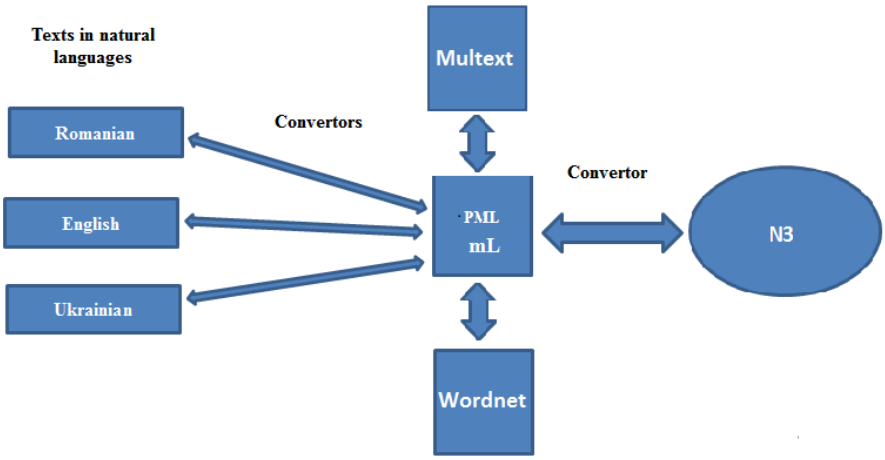


Figure 1. Semantic Web – natural languages „interface".

In order to materialize the idea of communication in natural language with Semantic Web, further research is needed on the (1) logic of mL, (2) interoperability of mL with WordNet as a source of English vocabulary or with a similar system for another language like Romanian, and (3) use of a morpho-lexical system like Multext (Tufiș *et al.*, 1998) to make use of the inflection paradigm. The research for the nearest future is on the logic of mL planned to be conducted in cooperation with Ukrainian researchers working, in particular, on the topics treated in (Nikitchenco, 2012).

5. Conclusions and discussion

The „proof of concept" programs showed an average degree of disambiguation for English texts higher than the statistical expectation, and for simple texts, this was significantly higher. The estimation was done by a human, and such programs were developed in Stanford NLP Java framework which is focused on English. This makes it difficult to properly evaluate the added value of this markup. But Romanian has a rich inflection paradigm, and inflections add extra information for disambiguation. Therefore, one can expect a higher average of correct disambiguation for Romanian. On the other hand, to „translate" Semantic Web

ontology to Romanian would be significantly more difficult than in English, and for the same reason.

But PML has an obvious limitation – it is totally insensitive to the mass-count distinction (Nicolas, 2008) and the disambiguation by use of PML can be trustworthy only if it is applied to texts which do not use mass-nouns. This limitation is due to the semantics of mL which is based on set theory – a theory of atoms, sets, ordered pairs and classes, all of which are countable objects in the sense used to make the mass-noun distinction.

References

Berners-Lee, T., Fischetti, Mark (2000). Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor. San Francisco: Harper.

Church, A. (1974). Set Theory with a Universal Set. Proceedings of the Tarski Symposium. American Mathematical Society. pp. 297-308.

Drugus, I. (2007). A Wholebrain approach to the Web. In *Proc. of Web Intelligence – Intelligent Agent Technology Conference*, Silicon Valley, pp. 68-71.

Drugus, I. (2009a). Universics: an Approach to Knowledge based on Set theory. In *Knowledge Engineering Principles and Techniques*. Selected Extended Papers. Cluj-Napoca, Romania, pp. 193-200.

Drugus, I. (2009b). Metalingua – a Formal Language for Integration of Disciplines via their Universes of Discourse. *Economy Transdisciplinarity Cognition Journal of Bacovia University*, Vol. 12, N2, pp. 17-23.

Drugus, I. (2010). Universics: a Common Formalization Framework for Brain Informatics and Semantic Web. In *Web Intelligence and Intelligent Agents. InTech Publishers*, Vucovar pp. 55-78.

Drugus, I. (2012). Metalingua: A Language to Mediate Communication with Semantic Web in Natural Languages. In *Advanced Information Technology in Education*, AISC 126. K.S. Thaung (Ed.), 2011, Springer-Verlag Berlin, Heidelberg, pp. 109–115.

Drugus, I. (2015). Universics: an Axiomatic Theory of Universes for the Foundations (in two parts). In *Proceedings of the Workshop on Foundations of Informatics*, August 24-29, 2015, Chișinău, Republic of Moldova. pp. 118-153.

Frege, G. (1892). On Sense and Reference. In Translations from the Philosophical Writings of Gottlob Frege. Blackwells, London (1892/1966). P. Geach, M. Black(Eds.). pp. 56-78.

Nikitchenko, M., Chentsov, A. (2012). Basics of Intensionalized Data: Presets, Sets, and Nominats. *The Computer Science Journal of Moldova* 20(3), pp. 334-365.

Nicolas, D. (2008). Mass Nouns and Plural Logic. Linguistics and Philosophy, 31(2), pp. 211–244.

Rogers, A. M. (2012). Cognitive set theory. ArborRythms, Boston, MA.

Quine, W. V. (1981). Mathematical Logic (Revised ed.). Cambridge, MA.

Tufiş, D., Ide, N., Erjavec, T. (1998). Standardised Specifications, Development and Assessment of Large Morpho-Lexical Resources for Six Central and Eastern European Languages. *First International Conference on Language Resources and Evaluation*, Granada, pp. 233-240.

THE "QUO VADIS" STORYTELLING

MIHAELA COLHON[1], DANIELA GÎFU[2], DAN CRISTEA[2,3]

[1] *Department of Computer Science, University of Craiova*
[2] *Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iaşi*
[3] *Institute for Computer Science, Romanian Academy – the Iaşi branch*
mcolhon@inf.ucv.ro, {daniela.gifu, dcristea}@info.uaic.ro

Abstract

Storytelling refers to finding the manners in which characters mentioned in a book are interconnected. In this study we are interested to go one step further in the semantic interpretation of a free text by exploiting the semantic relationships between major characters mentioned in the text. As material for our research, a heavily annotated Romanian translation of the well known "Quo Vadis" novel was used. Provided certain categories of semantic relations can be automatically deciphered, valuable information can be extracted about the text, among which: ranking the importance of characters in the novel, evolution of relations between major characters profiling the story, stamping of the empathy induced by different characters in readers on a positive-negative axis, and even elements of a general summary of the story.

Keywords: corpus, entities linking, semantic relations, storytelling.

1. Introduction

There have been many attempts to provide computational models for narrative and storytelling (Tapscott *et al.*, 2014). In this paper we present a study aimed at configuring features of a storytelling that uses the semantic relations annotated in a text. More precisely, we propose a model for visualising statistics about the "Quo Vadis"[1] novel. This model allows the organisation of semantic data extracted from the text by means of graph structures and diagrams. There are more types of semantic graphs that can be represented based on the semantic information extracted from a text. In such a representation, for instance, each node uniquely represents a character while the arcs represent semantic relations between pairs of characters, as are they mentioned in the text. Nodes and arcs can also be labeled with attributes depending on the knowledge domain (Sowa, 1984).

Our study is focused on affective, social and kinship relations. The proposed model inspires also the elaboration a query engine capable to answer queries about the

[1] Authored by Henryk Sienkiewicz, Polish writer, Nobel laureate in literature (1905).

characters of the novel and the way they interact. The approach represents one step further in the challenging task of automatic deciphering the content of free texts.

The paper is organized as follows. In the following section we give an overview of some studies developed in storytelling. In Section 3 we present the QuoVadis corpus. Section 4 describes the three schemes that make up our storytelling model and Section 5 gives insights about a query engine that would use the semantic knowledge annotated in the corpus. The last section gives concluding remarks and presents some future work.

2. Related work

Story generation, storytelling, and story understanding are examples of a phenomenon called *narrative intelligence* (Li *et al.*, 2013). The task of storytelling, also known as the problem of "connecting the dots", is already used in various contexts: entity networks (Fang *et al.*, 2011), image collections (Heath *et al.*, 2010), social networks (Faloutsos *et al.*, 2004) or document collections (Hossain *et al.*, 2012). These representations can be used to create connections between disparate entities described in text in order to discover relationships between different concepts, trying in this manner to suggest story-based hypotheses. For example, what is the connection between "Personage X" and "Personage Y"? What other characters relate them and what type of connections exist between them?

Many software tools were already developed to support storytelling (Eccles *et al.*, 2008; Winston, 1999, Kumar, 2008). From the diversity of tools developed in this space, are worth mentioning here the "Entity Workspace" (Bier *et al.*, 2006), which uses entity recognizers to infer a graph of relationships between entities. The MUSE project introduces a new way of exploring and understanding information by "bringing text to life" through 3D interactive storytelling (Winston, 1999). Endert *et al.* (2014) describe visual analytics approaches for exploring connections in document collections and for building stories between possibly disparate end-points. However, following these authors, sophisticated analytics support for storytelling remains a significant research frontier.

With the intended scope of developing a new language independent storytelling model, in this paper we present an Entity-Relationship model built upon the semantic relations manually annotated in a literature text.

3. The QuoVadis corpus

Cristea *et al.* (2015) describe a research aiming to build a corpus of semantic relations. The corpus, called QuoVadis, from the novel used as hub document, includes annotated entities and semantic relations.

Until now, several Web resources have been made available to the interested researchers:

- the QuoVadis corpus itself, an XML coded file[2], containing 7281 sentences annotated with semantic data;
- a Web interface for visualizing the coreference relations manually marked in the corpus[3];
- the collection of all the semantic relations extracted from the corpus, together with their syntactic patterns[4] (Bibiri *et al.*, 2014). This data was intended to develop an automatic recognizer for semantic relations.

In (Cristea *et al.*, 2015) the marking conventions of the QuoVadis corpus are presented. They include annotations for persons and god type entities, including groups, and for 4 types of relations linking them: referential, affect, kinship and social (the last 3 globally called AKS). The annotation itself was a time consuming and painful process, that run over more than two years. The annotators were master students in Computational Linguistics and a couple of PhD researchers in linguistics and computer science. The specifications initially developed had to be updated repeatedly to cope with the diversity of cases, until a satisfactorily fixed version was obtained. The reason for this was to leave less room for personal interpretations during the annotation process that would have made the markings rather subjective. The extremely high density of markings in the corpus made impossible to double the annotators' work and to organize a proper agreement exercise. Nevertheless, the quality of the corpus was surveyed by grouping the students in pairs and reciprocally sampling, reporting and correcting among them differences of views. The corpus was submitted to a number of tests and iterative corrections until a sufficiently high degree of accuracy was obtained.

With respect to the relative positioning in the text of the spans of text that realise the entities forming the two arguments of the relations, they could be intersectable or not. If they intersect, then (empirical evidence show that) they are necessarily nested (imbricated) and the convention for the direction of the relation is to consider on the *FROM* role the larger entity and on the *TO* role the nested entity[5] (Colhon *et al.*, 2016). Some examples are show below:

1:[<*copilul*> *drag* 2:[*al celebrului Aulus*]] (in the English version, 1:[*a dear* <*child*> *of* 2:[*the famous Aulus*]]); the *FROM* entity noted here with [1] includes the *TO* entity denoted by [2] and there is a `child-of` relation linking [1] and [2]. The relation is signalled by the word <*copilul*> (EN <*child*>), called trigger.

When entities are non-imbricated, the normal reading of the trigger usually gives the direction of the relation. The following example shows to what extend such annotations could be complex:

... 1:[*Poppea*], *iar de când* 2:[*i*]-*a* 3:[<*născut*>;REALISATION=INCLUDED] 4:[*o fiică*], 5:[*Nero*] *este și mai mult sub puterea* 6:[*ei*] ... (EN: *Here* 1:[*Poppea*] *rules;*

[2] http://nlptools.info.uaic.ro/tools
[3] http://nlptools.infoiasi.ro/QuoVadisVisualization/
[4] http://mcolhon.ro/qv/index.html
[5] Here and below, entities are marked in square brackets and triggers in angular brackets.

and 5:[*Nero*], *since* 3:[*she*] <*bore*> 2:[*him*], 4:[*a daughter*], *is even more than ever under* 6:[*her*] *influence* ...).

In this example, the Romanian equivalent for *she* (*ea*) is not realised in the text, therefore we say that it is “included” in the verb form (entity [3]). In this example, the following relations are marked:

- [3], [6] and [1] are coreferential, and [5] is coreferential with [2];
- a `parent-of` relation links entity [3] to [4], with the trigger <*născut*> (in English, <*bore*>). Moreover, [2] is also a `parent-of` [4], as the father.

The AKS relations are as follows:

- the AFFECT class includes 11 subtypes: `friend-of`, `fear-of`, `fear-to`, `love`, `loved-by`, `rec-love`, `hate`, `hated-by`, `upset-on`, `worship` and `worshiped-by`;

- the KINSHIP class includes 7 subtypes: `parent-of`, `child-of`, `sibling-of`, `nephew-of`, `spouse-of`, `concubine-of` and `unknown`.

The SOCIAL class includes 6 subtypes: `superior-of`, `inferior-of`, `colleague-of`, `opposite-to`, `in-cooperation-with` and `in-competition-with`.

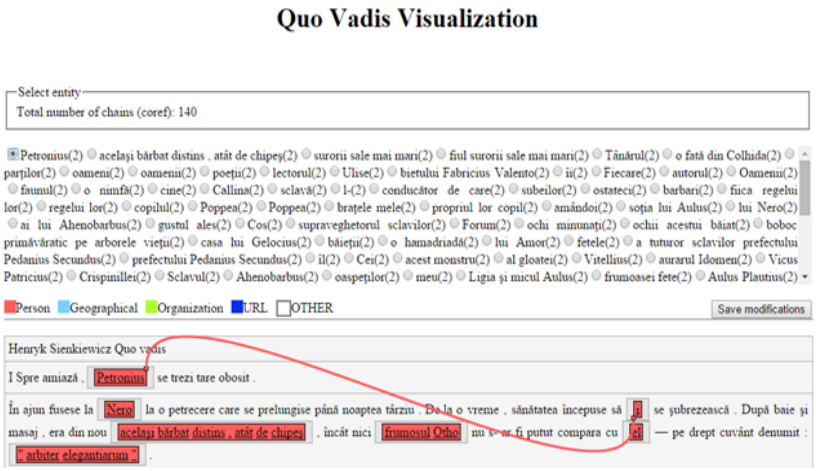


Figure 1. A print-screen of the QuoVadis Visualization Tool

4. Revealing the story

In this section we present a number of schemes that allow to make deductions from a semantic point of view over a literature text.

We call the first scheme *Entity-Centered* because it is built around the actors of the story and helps to assess the position of (mainly) central characters with respect to others. The scheme is presented in section 4.1. The second scheme, presented in Section 4.2, is called *Empathy-Centered*, and allows to detect the empathy on characters as felt by an average reader. Finally, the third scheme is called *Time-Ordered* and displays the relations in the text linearly, in the order of their mentions. "Time" appearing in the name of this scheme refers to the property of the text to have exactly one story threadthat is unfolded linearly in time.

As said already, the text under investigation is the novel "Quo Vadis", manually augmented with semantic annotation (Cristea *et al.*, 2015), but, at least for the first two schemes, the model can be applied to any other texts as well. However, the third scheme is restricted to texts observing a linear story pattern, as noticed.

The QuoVadis corpus includes two layers of markings: the lower layer, contributed automatically, including part of speech and morphological information and the higher layer, contributed manually, marking entities and the 4 types of semantic relations.

The corpus includes very scarce mentions of classes, the majority of entities being instances (Tapscott *et al.*, 2014). In view of a story representation, the entities represent the *actors of the story* (see in Figure 1 a screenshot showing some coreference relations linking different mentions of the same characters).

Coreference resolution is the art of determining when an entity mention in a text refers to another. As an example, mentions as *o fiică de rege* (EN: *a daughter of a king*) or *Ligiei* (*to Ligia*), as well as all the pronouns that refer *Ligia*, are represented under a unique [*Ligia*] entity (actually sharing the same ID in the XML representation).

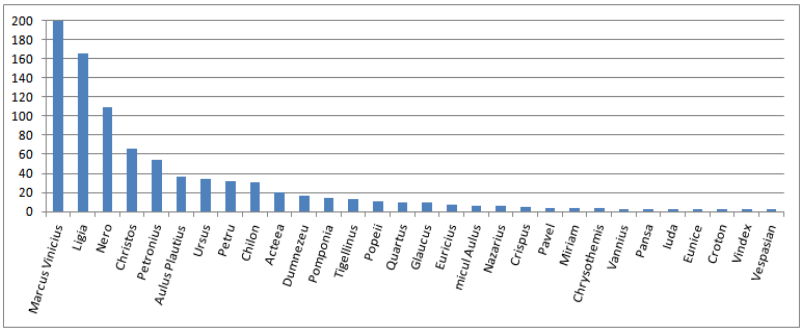


Figure 2. Ranking of characters in the descending order of the number of relations they are involved in (the first 30 characters)

4.1. Ranking characters and assessing their reciprocal relationships: the Entity-Centered scheme

The coreferential marking makes easy to deduce all semantic relations linking two characters in the novel, irrespective of the manner in which these characters are

being mentioned. Then, by simply counting the semantic relations that have the same character as an argument, a ranking of the salience of characters can be deduced. Therefore, *main characters* or the *main actors* of the story are the most often characters referenced (or involved in relations) - see Figure 2 above.

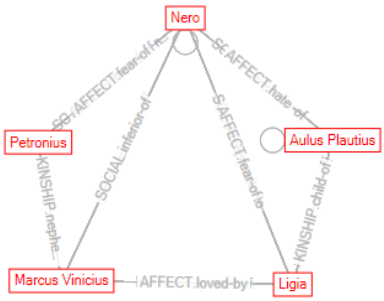


Figure 3. The main story in “Quo Vadis”

Figure 3 shows a semantic graph including nodes for the 5 most salient human characters in the novel and edges depicting the most frequently mentioned relations among them. The graph with the VIP characters on the nodes could be considered to give a graphical expression of one facet of the *main story*, because in the interpretation of a text it is important to figure out what kind of relationships are developed among the main characters.

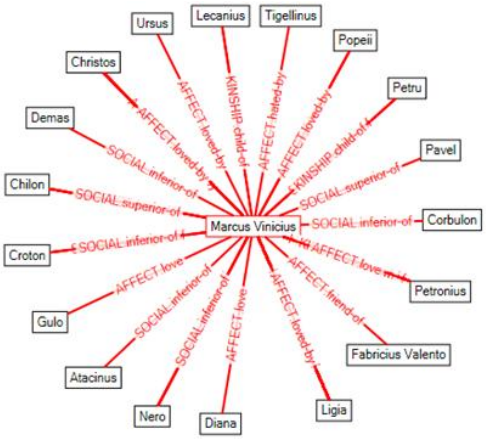


Figure 4. The “sub-story” of *Marcus Vinicius*

Moreover, each main character has its personal gallery of characters that connect to her/him. Figure 4 displays *Marcus Vinicius*’ star of connections, while Figure 5 displays the individual gallery of interconnections for *Aulus Plautius*, *Petronius*, *Nero* and *Ligia*. One way to look at these graphs is as *sub-stories* (or *satellite stories*) of the main story.

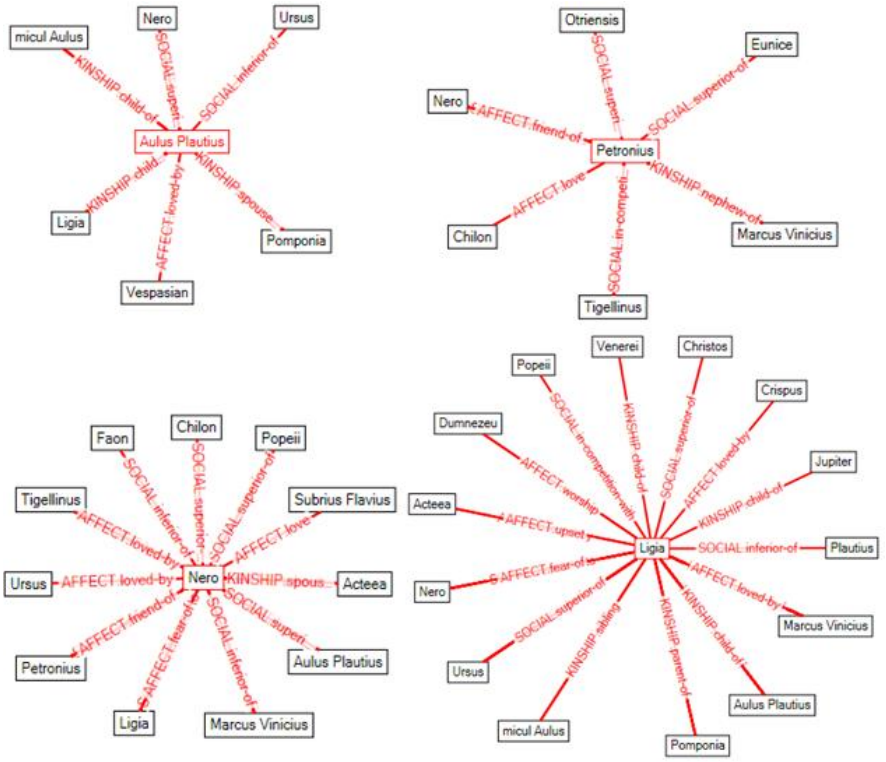


Figure 5. The sub-stories of *Aulus Plautius*, *Petronius*, *Nero* and *Ligia*

4.2 Appreciating the positive/negative empathy of readers towards characters: the Empathy-Centered scheme

Very often, a book character is complex, alternating between light and shadow, between good and evil. This triggers the change of the reader's empathy addressing her/him, as the story unfolds and new facts or events are accumulated, but at the end of the story a generic average opinion is usually stabilised. Although this valuation (happening unconsciously in readers' mind) is dependent on their idiosyncrasies, in many cases a kind of average opinion shows up statistically.

We have tried to simulate the empathy raised by characters in readers solely based on the characters' participation in semantic relations. In order to do that, we have asked a class of students to give scores to the roles *FROM* and *TO* of a number of representative relations (the participants in a relation *rel* are written as: *FROM rel TO*). The scores suggested by us were in the range: -3 to +3, where: -3 means "very negative", 0 means "neither negative, nor positive", and +3 means "very positive". We have left on purpose in the list, although redundant, some relations together with their inverses, in order to verify the attention of students.

The idea was that if *rel* and rel^{-1} are inverse relations (in our set these were: *love* and *loved-by*, *worship* and *worshiped-by*, *fear-of* and *fear-to*), then the valuation of the two arguments should be symmetrical in the two relations, i.e. if "*X1 rel Y1*" and "*X2* rel^{-1} *Y2*" then valuation(*X1*) = valuation(*Y2*) and valuation(*Y1*) = valuation(*X2*). We have eliminated from the set all records of students who have not passed this test for all three pairs. After doing this pruning, we still remained with a set of 89 records that we considered being valid. Figure 6 shows the diagrams of the valuations given by students to roles *FROM* and *TO* (in different colours, on each diagram). A proof that the set has no evident skews, as given by misunderstanding the task, is that symmetrical relations (such as *colleague-of* and *reciprocal-love*) indeed have identical shapes on the roles *FROM* and *TO*.

Our hope was that all these shapes resemble the Gauss curve, which, as seen, is not always the case. Curves that have two maxima, such as the inverses: *fear-of* and *fear-to* reflect two personalities of subjects: one that believes that having fear of somebody is a very negative feature and one that thinks the fearing persons are not to be blamed. More curious is that the person inspiring fear (the role *TO* in *fear-of* and the role *FROM* in *fear-to*) display almost similar shapes with the person experiencing fear.

Table 1: The positiveness/negativeness of roles of relations

Relation Type	**Relation Subtype**	**FROM Median Positivity (P) and Negativity (N) scores**		**TO Median Positivity (P) and Negativity (N) scores**	
SOCIAL	*colleague-of*	0.74 P	-	0.71 P	-
	opposite-to	-	1.92 N	-	0.54 N
	in-cooperation-with	1.51 P	-	1.21 P	-
	in-competition-with	-	0.62 N	-	0.43 N
AFFECT	*friend-of*	2.18 P	-	0.94 P	-
	love	2.43 P	-	1.02 P	-
	loved-by	1.02 P	-	2.43 P	-
	rec-love	2.53 P	-	2.62 P	-
	worship	1.82 P	-	1.52 P	-
	worshiped-by	1.52 P	-	1.82 P	-

	fear-of	-	1.20 N	-	1.31 N
	fear-to	-	1.30 N	-	1.26 N
	hate	-	2.31 N	-	0.93 N
	hated-by	-	0.93 N	-	2.31 N

Considering the median values of the plots given in Figure 6, we have grouped the roles of semantic relations in two categories: positive and negative. As such, Table 1 shows the positiveness/negativeness of the roles of all relations. As seen, actually all relations have both roles of the same polarity, which makes us believe that this is a property of the relation, more than of the role.

Within this assumption, for each character, two scores can be computed: the *positivity score* and the *negativity score* based on the positive, respectively negative, relations in whom she/he is involved along the story. We introduce some notations first:

$$\text{POS-RELS} = \{R \mid 0 \leq SC_R^{FROM} \leq 3,\ 0 \leq SC_R^{TO} \leq 3\}$$

$$\text{NEG-RELS} = \{R \mid -3 \leq SC_R^{FROM} < 0,\ -3 \leq SC_R^{TO} < 0\}$$

where R is a relation and SC_R^{FROM} and SC_R^{TO} are the medians of the histograms of scores (in Fig. 6) of the two poles of this relation.

These two sets of relations being set, we can define the participation of characters in relations as:

$$\text{POS-RELS}(E) = \{R_i^E \mid R_i^E \in \text{POS-RELS}\}$$

$$\text{NEG-RELS}(E) = \{R_i^E \mid R_i^E \in \text{NEG-RELS}\}$$

where R_i^E is an occurrence of a relation R in which E appears on any of the roles *FROM* or *TO* (actually expressions like $R_i^E \in$ POS-RELS should be read as: the instance i of the relation R having on one of the roles the entity E is in the set POS-RELS). Moreover, we will mark as $SC_{R\_i}(E)$ the median score of the occurrences R_i^E.

With this notation, we may now define the average scores:

$$SC_{POS}(E) = \frac{\sum_{R\_i \in POS\_RELS(E)} SC_{R\_i}(E)}{|RELS(E)|},\ SC_{NEG}(E) = \frac{\sum_{R\_i \in NEG\_RELS(E)} SC_{R\_i}(E)}{|RELS(E)|}$$

which can be used to compare the overall positive and negative behaviour of a character. Table 2 shows a comparison, on some of the key characters of the novel, between the interpretation given by one of authors of this paper and the interpretation computed according to the scores above. Since, the average scores on the 4th column do not show the quantity of relations involved, we thought useful to place also the sum scores (the last column).

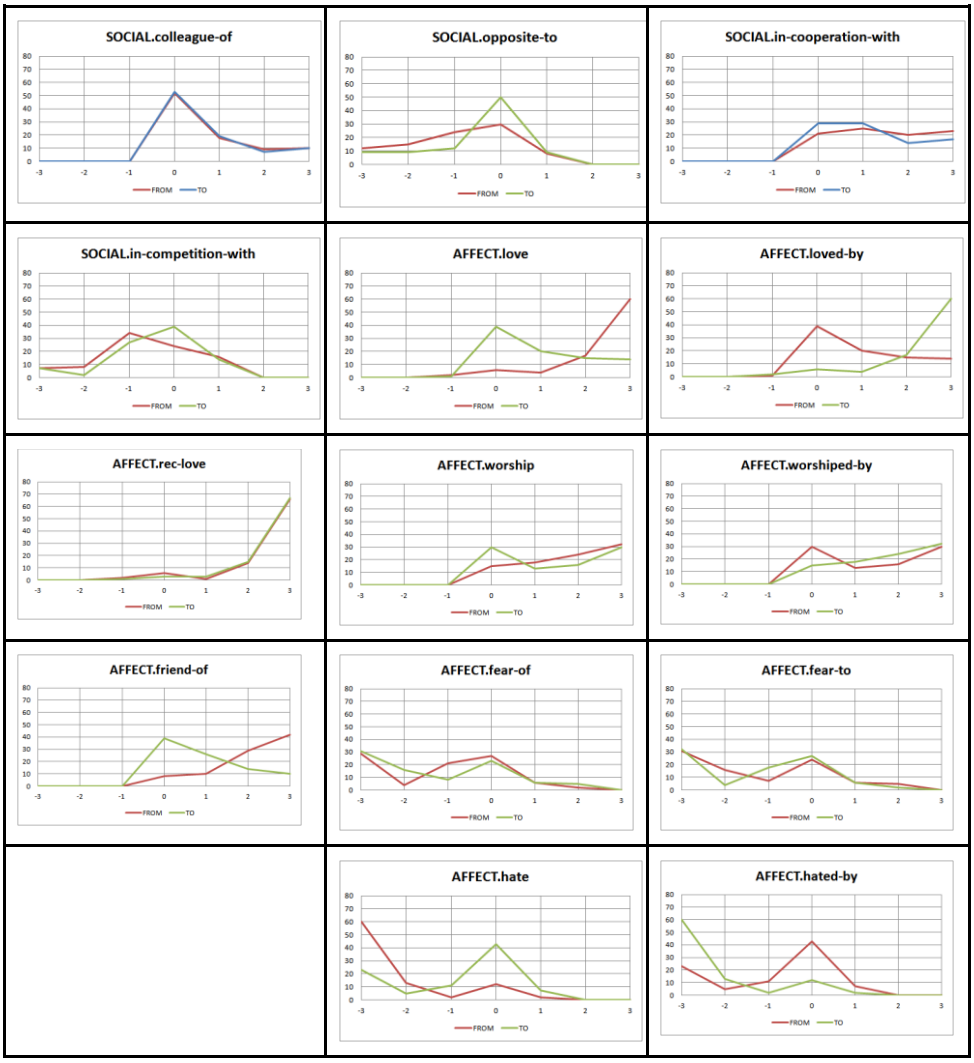


Figure 6. Diagrams showing the valuation of roles *FROM* and *TO* for the semantic relations used in the test

It can be noticed that interpretation of the empathy raised by characters in readers based solely on relations’ roles of characters seems to be balanced towards positive interpretations.

For instance, Nero is seen as a positive character and receives equal scores with Petronius. We put this tendency on the fact that many characters express false appreciations towards their emperor, positions which are impossible to be revealed with our method.

Another example is Chilon, which has an ambiguous position in the book, partly positive, partly negative. As for Popeea, she is perceived negative due to her closeness to Nero, rather than by evaluations stated by other characters.

Table 2. The positivity/negativity empathy regarding the first 14 most salient characters in "Quo Vadis": judged by a good connoisseur of the novel against the ones computed automatically

No.	Characters	Positivity/ negativity feature of the character, as expressed by an expert	Positivity/negativity of the character expressed as averages: $SC_{POS}(E)$ $SC_{NEG}(E)$	Positivity/negativity of the character expressed as sums: $\sum_{R\_i \in POS\_RELS(E)} SC_{R\_i}($ $\sum_{R\_i \in NEG\_RELS(E)} SC_{R\_i}($
1.	Marcus Vinicius	P	**1.64 P**; 0.15 N	**152.55 P**; 13.95 N
2.	Ligia	P	**1.35 P**; 0.19 N	**97.55 P**; 13.5 N
3.	Nero	N	**0.95 P**; 0.40 N	**49.77 P**; 20.74 N
4.	Christos	P	**1.31 P**; 0.16 N	**34.01 P**; 4.06 N
5.	Petronius	P	**0.94 P**; 0.46 N	**29.0 P**; 14.4 N
6.	Aulus Plautius	P	**1.27 P**; 0.32 N	**7.61 P**; 1.93 N
7.	Ursus	P	**1.25 P**; 0.29 N	**16.24 P**; 3.82 N
8.	Petru	P	**1.39 P**; -	**4.16 P**; -
9.	Chilon	N,P	**1.13 P**; 0.37 N	**18.16 P**; 5.97 N
10.	Acteea	P	**1.74 P**; 0.36 N	**15.70 P**; 3.23 N
11.	Pomponia	P	**0.95 P**; 0.48 N	**3.78 P**; 1.93 N
12.	Tigellinus	N	0.56 P; **1.07 N**	5.63 P; **10.72 N**
13.	Popeea	N	**0.81 P**; 0.65 N	**4.86 P**; 3.9 N
14.	Glaucus	P	0.81 P; **0.95 N**	2.43 P; **2.85 N**

The conclusion is still insecure, because of the scarcity of relations expressed in the novel with respect to negative characters.

4.3 Seeing relations in time. The Time-Ordered scheme

A story is an ordered sequence of series of events that are related in some way. The traditional stories’ order of events roughly corresponds with time, which is crucial to understand causality (Austin, 2011): events that happen earlier can influence later events, but not the other way round. The QuoVadis corpus does not include temporal annotation.

However, temporal data can be inferred from the approximately linear unfolding of the story over time, which is characteristic in the novel under investigation. As such, the temporal order can be mapped on the offset of words (or sentences). In this section we present a text characterising scheme which relies on this mapping. It is still important to mention that this mapping should not be taken as a general rule. Modern novels very often departure from the linear pattern of story unfolding, the authors obliging the readers to truly zigzagging back and forth in the story time, by flashbacks and prefiguring events belonging to the future.

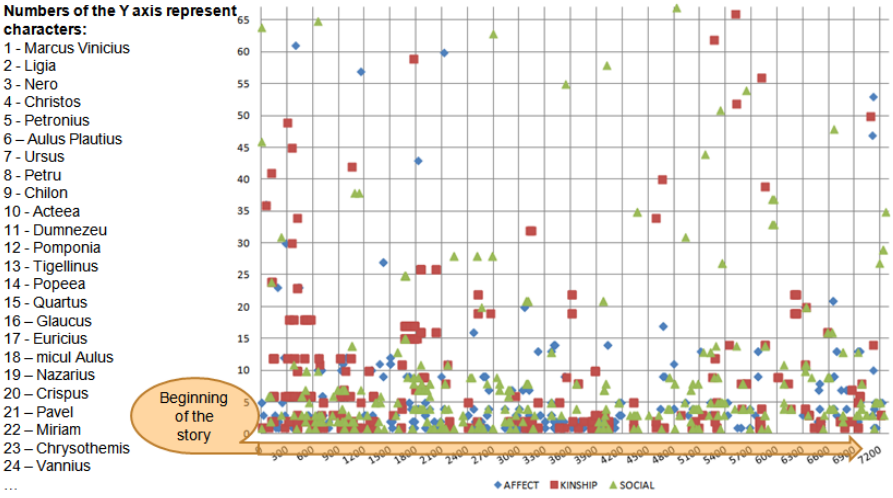


Figure 7. A timestamp of the involvement of “Quo Vadis” characters as agents in AKS relations, more salient low, less salience up

In Figure 7 we display a timestamp of the relations the main characters are involved in in the novel. The OX axis represents the numerical IDs of sentences in the text (in ascending and consecutive order), and the OY axis displays characters, the most important at the bottom, the less important - at the top. A dot in this representation signifies the occurrence of the Y character in a semantic relation on the *FROM* role, in the X^{th} sentence of the book.

It is clear that representations of this kind could be exploited in different ways, for instance, visualising only a selection of characters, or a subset of relations, or filtering only relations between two characters.

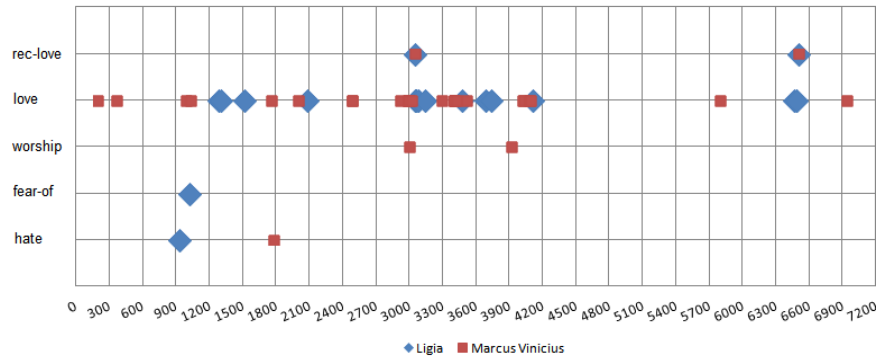


Figure 8. The development of the affective relationships between *Marcus Vincius* and *Ligia*

Following this idea, the diagram in Figure 8 puts in evidence the development of the affective relationships between *Marcus Vinicius* and *Ligia*. The diagram shows that Marcus is more inclined in declaring his love to Ligia than vice-versa. However, shortly after making acquaintance, but after Marcus has fallen in love with Ligia, she experienced first hate and then fear towards Marcus. Her sentiments evolved into love later on. There is also a moment in the story development (at about sentence 1800) when Marcus feels almost simultaneously both hate and love towards Ligia, showing the vexation and storm of his soul.

5. Expressing queries

The schemes presented in section 4 are intended to provide knowledge data for a query engine. But this approach suggests further a processing mechanism adequate for reasoning about characters, about their involvement in events and places, or that could be used to provide answers to complex questions about the story.

The relationships marked in the corpus can be used to feed a database of facts for a query engine. Facts are given in *subject-verb-object* statements that involve entities which, conforming to annotations in the corpus, are described by quadruples:

(sentence ID, relationship, the *FROM* entity, the *TO* entity)

The underscore (_) on the position of sentence ID means ANY.

In what follows we give examples of possible queries:

- the family members of a character *X*: gather all instances *Y* that are involved in KINSHIP relations with X

```
select all Y: (_,KINSHIP.<any_subtype>,Y,X) OR
(_,KINSHIP.<any_subtype>,X,Y)
```

- the friends of a character *X*: consider all *AFFECT.friend-of* relations of *X* in a symmetric manner

```
select all Y: (_,AFFECT.friend_of,X,Y) OR
(_,AFFECT.friend_of,Y,X)
```

- the best friends of a character *X* can be defined as characters involved in a maximum number of *AFFECT.friend-of* relations with *X*

```
select all Y: max of
count((_,AFFECT.friend_of,X,Y) OR
(_,AFFECT.friend_of,Y,X))
```

- the superiors of a character *X* on a social scale: all entities that are in *SOCIAL.superior-of* relations with *X* in a transitive chain

```
select all Y: (_,SOCIAL.superior-of,Y,X) OR (there
is Z such that (_,SOCIAL.superior-of,Y,Z) AND
(_,SOCIAL.superior-of,Z,X))
```

- the evolution of sentiments between two characters X and Y

```
select all t1,t2: (t1,AFFECT.hate,X,Y) AND
(t2,AFFECT.love,X,Y)
```

Moreover, a number of new facts can be derived via more or less certain implications (by a kind of “weak transitivity” property) of the form “if *X rel1 Y* and *Y rel2 Z* then *X rel3 Z* with a certain probability”, where *X*, *Y*, and *Z* are entities and *rel1*, *rel2*, and *rel3* are some AKS relationships.

Examples of such triplets of relationships and their estimated probabilities are: (*parent-of*, *parent-of*, *grandparent-of*, *1*), (*superior-of*, *superior-of*, *superior-of*, *1*), (*in-competition-with*, *friend-of*, *enemy-with*, 0.8), (*friend-of*, *in-competition-with*, *enemy-with*, 0.8). Here are some examples:

- grandparents of a character X: the directly expressed grandparents together with the parents of her/his parents

```
select all Y: (_,KINSHIP.grandparent-of,Y,X) OR
(there is Z such that (_,KINSHIP.parent-of,Y,Z)
AND (_,KINSHIP.parent-of,Z,X))
```

- possible enemies of a character *X*: the directly expressed opponents of *X* together with all their friends and with the opponents of *X*’s friends

```
select all Y: (_,SOCIAL.in_competition_with,X,Y)
OR (there is Z such that
((_,SOCIAL.in_competition_with,X,Z)AND
(_,AFFECT.friend-of,Z,Y))) OR (there is Z such
that ((_,AFFECT.friend-of,X,Z)
AND(_,SOCIAL.in_competition_with,Z,Y)))
```

6. Conclusions and future work

Summarising, the proposed Entity-Relationship storytelling model can offer help to someone interested in criticizing about a novel. Even making use of a very restrained set of relations (as are those annotated in our corpus), the model still opens a number of possibilities of reasoning about characters and their interrelationships. For example, it can offer arguments for ranking the characters in the order of their importance, show the way they interrelate, in both a static manner (as types) or dynamic (on the time axis), and help in configuring a point of view with respect to their positivity/negativity.

The model can be augmented in many respects, before configuring a technology able to really "understand" a literature text. Among these, we mention: a) the automatic recognition of entities and relationships in free texts; b) the improvement of the anaphora resolution technique, since errors in the coreference chains most of the time trigger false conclusion; c) the inclusion of temporal annotation, that would allow effective reasoning about time by changing the sentence ID axis onto a real time axis in the Time-Ordered scheme; d) the recognition of events, not only of semantic relations, and visualisation of stories (van Erp *et al.*, 2014). One of the applications that can be based on our model is summaries drafting for large texts, since the big lengths of novels make impossible the classical extract techniques (that reduce the size down to a certain percentage). Instead, summarising long texts involves: recognition of main characters, their genders, ages, professions and kinship relationships, deciphering of the intriguing events or situations they are involved in, and a short time reconstruction of the story. Ready-made queries of the types exemplified in Section 5 can be used to profile interesting links between characters or relieve surprising evolutions of the story.

Acknowledgements

The study reported in this paper was supported by the MappingBooks project (code: PN-II-PT-PCCA-2013-4-187). The QuoVadis corpus is part of CoRoLa (the Computational Corpus of Contemporary Romanian Language), under development by the Romanian Academy.

References

Austin, M. (2011). Useful Fictions: Evolution, Anxiety, and the Origins of Literature. University of Nebraska Press.

Bibiri, A.-D., Colhon, M., Diac, P., Cristea, D. (2014). Statistics over a Corpus of Semantic Links: "QuoVadis". In Colhon, M., Iftene, A., Barbu Mititelu, V., Cristea, D., Tufiş, D. (eds.) *Proceedings of the 10th International Conference "Linguistic Resources And Tools For Processing The Romanian Language"*, Craiova, 18-19 September 2014, „Alexandru Ioan Cuza" University Publishing House, pp. 33-44.

Bier, E, Ishak, E., Chi, E. (2006). Entity Workspace: An Evidence File That Aids Memory, Inference, and Reading. In *Proceedings of ISI ’06*, 2006, pp. 466–472.

Colhon, M., Cristea, D., Gîfu, D. (to appear in 2016). Discovering Semantic Relations within Nominals. In D. Trandabăț and D. Gîfu (eds.): *Proceedings of the Workshop on Social Media and the Web of Linked Data*, *RUMOUR-2015*, A satellite event of EUROLAN-2015, Sibiu, Romania, July 2015, Springer International Publishing.

Cristea, D., Gîfu, D., Colhon, M., Diac, P., Bibiri, A.-D., Mărănduc, C., Scutelnicu, L.-A. (2015). Quo Vadis: A Corpus of Entities and Relations. In N. Gala, R. Rapp and G.B. Enguix (eds.): *Language Production, Cognition, and the Lexicon*, Springer International Publishing Switzerland, vol. 48, pp. 505-543.

Eccles, R., Kapler, T., Harper, R., Wright, W. (2008). Stories in GeoTime. In *Info. Vis.* vol. 7, no. 1, pp. 3–17.

Endert, A.,Shahriar Hossain, M., Ramakrishnan, N., North, C., Fiaux, P., Andrews, C. (2014). The human is the loop: new directions for visual analytics. In *J Intell Inf Syst*, Springer, New York.

Fang, L., Sarma, A. D., Yu, C., Bohannon, P. (2011). Rex: explaining relationships between entity pairs, in *Proc. VLDB Endow.*, vol. 5, no. 3, pp. 241–252.

Heath, K., Gelfand, N., Ovsjanikov, M., Aanjaneya, M., Guibas, L. (2010). Image Webs: Computing and Exploiting Connectivity in Image Collections. In *CVPR*.

Hossain, M. S., Gresock, J., Edmonds, Y., Helm, R., Potts, M., Ramakrishnan, N. (2012). Connecting the Dots between PubMed Abstracts. In *PLoS ONE*, vol. 7, no. 1.

Faloutsos, C., McCurley, K. S., Tomkins, A. (2004). Fast Discovery of Connection Subgraphs. In *KDD ’04*.

Kumar, D., Ramakrishnan, N., Helm R.F., Potts, M. (2008) Algorithms for storytelling. In *IEEE Trans. Knowl. Data Eng*. vol. 20, pp. 736–751.

Li, B., Lee-Urban, S., Johnston, G., Riedl, M. O. (2013). Story generation with crowd-sourced plot graphs. In: *The 27th AAAI Conference on Artificial Intelligence*.

Tapscott, A., Colàs, J., Moghnieh, A., Blat, J. (2014) Modifying Entity Relationship Models for Collaborative Fiction Planning and its Impact on Potential Authors, in *Proceedings of 5th Workshop on Computational Models of Narrative* (CMN’14). Editors: Mark A. Finlayson, Jan Christoph Meister, and Emile G. Bruneau; pp. 209-221.

Sowa, J. F. (1984). Conceptual Structures: Information Processing in Mind and Machine. Reading, MA: Addison-Wesley.

van Erp, M., M. Nijsen, G. Satyukov, P. Vossen (2014). Discovering and visualising stories in news, in *Proceedings of the 9th Language Resources and Evaluation Conference* (LREC2014), 26-31 May, 2014, Reykjavik, Iceland.

Winston, E. A. (1999). Storytelling and Conversation: Discourse in Deaf Communities. Washington: Gallaudet University Press, 1999.

CHAPTER 4
TEXT ANALYTICS

A LEXICAL DISCOURSE ANALYSIS FRAMEWORK

IULIANA-MARIANA BEJAN, ADRIAN IFTENE, DANIELA GÎFU

Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iași

{iuliana.bejan, adiftene, daniela.gifu}@info.iasi.ro

Abstract

This paper examines the automatic classification of discourse types in written language data. Our goal was to implement an application that provides to users the grammatical analysis of content, word wrapping in a speech category, identifying potential mistakes related to spelling or expressing in order to facilitate the proper categorization of the discourse depending on the text category. We hypothesize that we can recognize the discourse type (here we use four kinds of texts on medical, economic, political and religious issues) as well as avoid some errors using a trained Naïve Bayes classifier. This approach can be useful to the direct beneficiaries (PR staffs from four areas: political, economic, religious and medical area), but, also, to specialists in natural language processing, linguists, etc.

Keywords: discourse classification, spellchecking, data pre-processing, Naïve Bayes.

1. Introduction

This paper describes the analysis and interpretation of different discourse types for Romanian language. For this purpose, we developed an application for grammatical analysis of the content, performing text categorization, identifying errors related to grammatical structure of the text, spelling, or expression, in order to facilitate proper drafting of the speech by the category it belongs to. Moreover, in this study, we propose several techniques that could be used to correct possible errors that might be encountered in a speech.

The analysis and interpretation of various types of speeches is still a challenge in NLP (*Natural Language Processing*) area. The complexity of this task is caused by multiple criteria that can be considered in the classification process. Practically, their coverage in an ideal scheme is impossible. Although the discourses' diversity is very high, we remember one of the most popular typology of speech, in three classes, *enunciative* (Benveniste, 1966), *communicational* and *situational* (Petitjean, 1989) which clarify the notion of discursive formation necessary in the discourse analysis (Pêcheux, 1990). In this context, only a part of what can be said is accessible, forming a system and defining an identity (Maingueneau, 1984).

Discourse analysis is a complex process which aims to establish the essence of a discourse, identifying its specific features and those of each individual (e.g. grammar structure, semantic identification, verifying the structural correctness and collecting this information to be used later in the interpretation process).

The paper is structured as follows: section 2 presents a selection of similar work and literature we relied on for pursuing our study, section 3 describes our system architecture, and section 4 details pre-processing phases. Finally, the conclusions and some directions for future work are addressed in section 5.

2. Background

Discourse analysis can be conceptualized both as a general method and, in a concrete way, as a cluster method or a field research. In other words, we speak about a reading and examination process. The following are examples of a series of computer applications developed for similar goals: Tropes[1], RO-Balie (Frunză *et al.*, 2005), and other competing software as Gate[2], Oak[3] or Minor Third[4], that provide the same services as RO-Balie[5], except RO-Balie could be trained, using machine learning techniques.

Tropes V1.0, developed in 1994, is the first text analysis software based on "propositional analysis principles: discourses are cut in propositions (simple phrases), considered as micro-universes concentrating a simple and self-sufficient meaning" (Grivel and Bousquet, 2011). It is available in two versions: French and English (as of November 7, 2011) Tropes is a semantic text analysis software, creating a specific thesaurus for a certain domain, allowing to perform a morpho-syntactic analysis on the text by identifying the morphologic category of all words, including: nouns, verbs, adjectives, pronouns, connectors, etc.

The lexico-semantic analysis phase, one of the most complex parts of Tropes, is used when analysing: the literary, philosophic, politic and scientific discourse, the semantic structure for portals and encyclopaedias, etc.

The similarities between Tropes and our tool are that both perform a morpho-syntactic analysis on the text, identifying the main part of speech, and both could be used to analyse different types of discourse such as: political discourses.

RO-Balie is an extension of BALIE[6], known as a multilingual text processing tool designed to support information extraction. The tool supports five languages: French, German, Spanish, English, and Romanian. It offers the following services: language identification, text segmentation in sentences, tokenization, and part-of-speech tagging. Balie text processing consists of a structured and rich list of tokens,

[1] Tropes Text analysis software, designed for TextMining, Qualitative Analysis, Semantic Categorization and Keywords extraction. In 1997, Acetic launches Tropes V3.0 automates the ACD (remarkable phrases), and chronological analysis of the text, starting from the political texts (http://www.semantic-knowledge.com/tropes.htm).
[2] http://gate.ac.uk
[3] http://nlp.cs.nyu.edu/oak
[4] http://minorthird.sourceforge.net
[5] RO-BALIE is available for download at http://www.site.uottawa.ca/~ofrunza/ RO-Balie/RO-Balie.html
[6] BALIE is a Java open-source software issued under the GNU General Public License. It is hosted by SourceForge and it is available at http://balie.sourceforge.net

and operates by applying machine learning techniques (Petijean, 1989). Both, RO-Balie and our tool allow to extract the tokens from the text and identify the main part-of speech.

Another similarity between RO-Balie and our framework is that both use the Naïve Bayes classifier: RO-Balie - for the language identification process, whereas, our tool uses it to identify the discourse type.

The most important difference between our tool and Tropes or RO-Balie, is that our tool, before performing any other action, it does a spellchecking on the text by identifying the errors and correcting them. It also displays these results in a table form allowing their export for later access.

3. System architecture

Developed entirely in Java, the web application allows users to upload, open and view files that make the subject of a linguistic analysis in order to determine the defining characteristics of the type of speech that it represents.

The application was designed to analyse Romanian texts, focusing on identifying the context of the analysed text, four areas being targeted: political, economic, religious and medical area.

In order to create the data that will be used as the training set for each type of speech, it was necessary to collect multiple texts from different sources, so that the extracted information will be diverse:

- sources for medical discourse were taken from the websites: http://www.stirimedicale.ro and "http://www.viata-medicala.ro",
- those for the economic discourse were taken from: "http://www.bnr.ro/Home.aspx", from the "Publications" section,
- those for the political discourse were extracted from: http://cms.presidency.ro/?pag=67 and http://www.tituscorlatean.ro/titus-corlatean-la-lansarea-candidatilor-usl-sector-1-bucuresti-pentru-alegerile-parlamentare-din-9-decembrie-2012/
- sources for the religious discourse were extracted from: http://www.ortodoxism.com/ and http://www.toaca.md/?&ziar_id=129, from the "News and Publications" section.

To use the program, users can either log in into the application in order to have access to all of the analysis tools made available through the program or to register so that they can later log in into the app.

Once authenticated into the app, a user can analyse a text choosing one of the three options: he may introduce a new text, or he may use an already saved text, or he may upload a new file, as it described in Figure 1.

The application allows analysis of files whose extension is text (.txt) and whose content is in Romanian. At the same time, the program allows both entering texts with diacritics and without diacritics. In the second case, the lack of diacritics will

be reported by the spellchecking process, the stage when the user will be given advice on how to replace the wrong words with their corresponding correct form.

As we mentioned in the beginning, the discourse analysis is a wide and complex area, so, we concentrated our attention only on four types of discourses: political, economic, medical and religious. For the moment, we have chosen only four categories of texts that can be analysed with the program because, over the years, these discourses have revealed a series of controversies, being four of the most common types of speech.

Subsequently, the program can be easily improved so that it can be used to analyse other types of discourses than the ones mentioned. This can be easily achieved by adding training data for other domains, data which will be used later in the analysis process.

The application is structured in four main modules, divided as follows: the module used to identify misspelled words (spellchecking) and provide suggestions for correcting them, the part of speech identification module, the module for identifying the type of the discourse and the module for displaying the results in form of statistics.

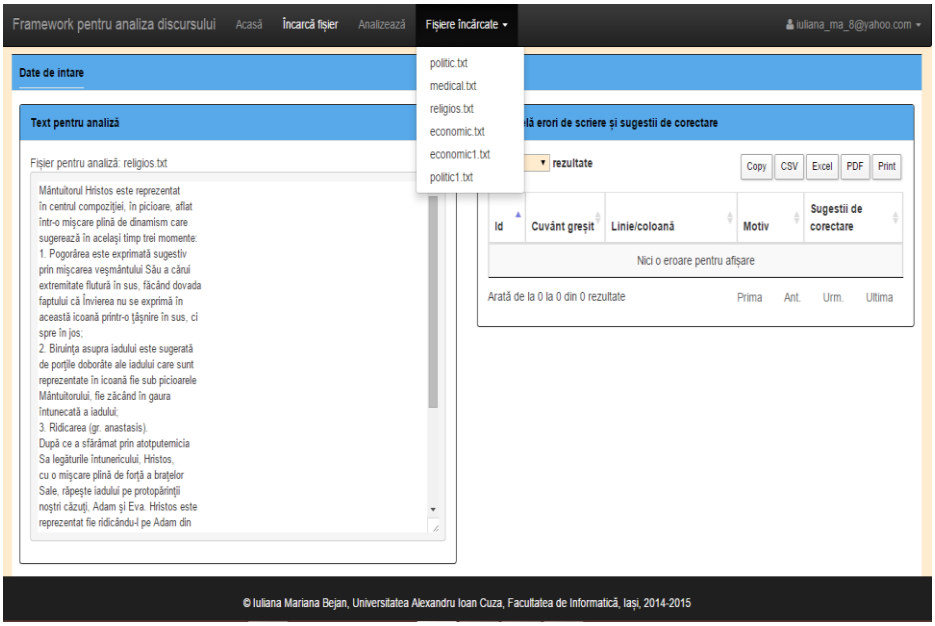


Figure 1. The interface after the correct authentication

3.1. Spellchecking

This module aims to identify errors in spelling or grammar, repeated words, and also identify if there is no punctuation or it is used incorrectly. This was done by integrating into the application the open-source program LanguageTool[7].

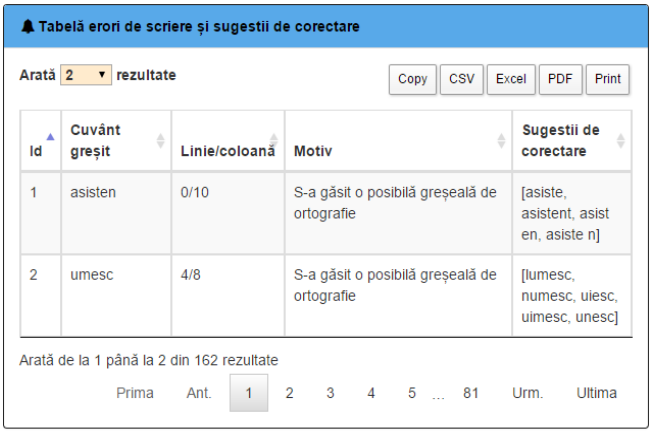


Figure 2. Spellchecking result table.

At the beginning, after the text was uploaded or after it was inserted, the program verifies if it contains any error and also identifies them. The results of this process are then displayed into a table consisting of: the possible mistaken word found by the program, the line and the column in the text where the word was identified and the description of the identified error, followed by a list of suggestions consisting of words that can be used to correct the identified term. Displaying the results into an error table makes data to be more compact and easy to follow. Also, by using properly the layout of the table, which offers the possibility to sort the results by any column of the table in ascending or descending order, the program allows users to easily browse through the results and to easily identify data in the table, see Figure 2.

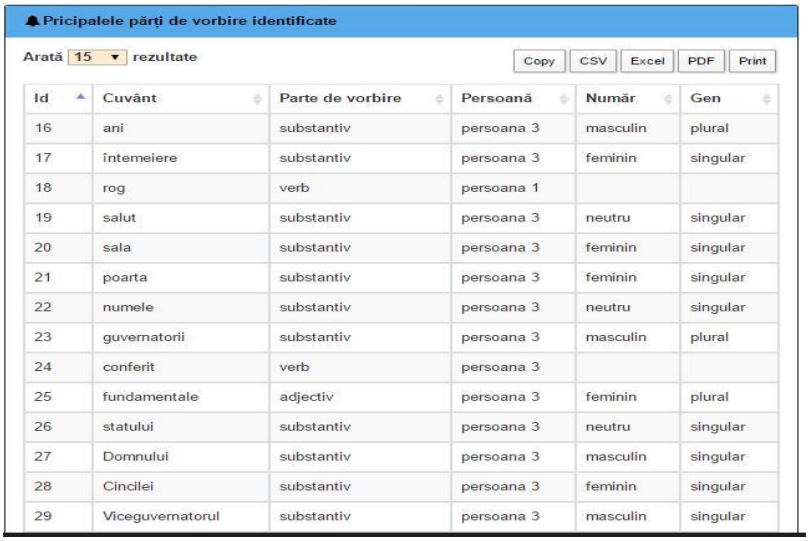
These results may also be exported as an excel document, a PDF document or a CSV file so that they can be reused later as reference for correct drafting of different texts.

3.2. Discourse POS-tagging

The second component of the application, involves identifying parts of speech associated with each key word from the text. In order to achieve this, we used the POS-tagger tool from the Language Tool library. This tool identifies for the most frequently used words in a document the corresponding part of speech. The obtained results forms a list consisting of different variations of the given word, each of them

[7] http://mvnrepository.com/artifact/org.languagetool/languagetool-core

followed by the abbreviation and the appropriate characteristics of the part of speech: person, number, gender, for nouns and adjectives, or just person, and, time, for verbs.



Pricipalele părți de vorbire identificate

Arată 15 rezultate

Copy CSV Excel PDF Print

Id	Cuvânt	Parte de vorbire	Persoană	Număr	Gen
16	ani	substantiv	persoana 3	masculin	plural
17	întemeiere	substantiv	persoana 3	feminin	singular
18	rog	verb	persoana 1		
19	salut	substantiv	persoana 3	neutru	singular
20	sala	substantiv	persoana 3	feminin	singular
21	poarta	substantiv	persoana 3	feminin	singular
22	numele	substantiv	persoana 3	neutru	singular
23	guvernatorii	substantiv	persoana 3	masculin	plural
24	conferit	verb	persoana 3		
25	fundamentale	adjectiv	persoana 3	feminin	plural
26	statului	substantiv	persoana 3	neutru	singular
27	Domnului	substantiv	persoana 3	masculin	singular
28	Cincilei	substantiv	persoana 3	feminin	singular
29	Viceguvernatorul	substantiv	persoana 3	masculin	singular

Figure 3. Table of identified part of speech

For example:

```
cer[cer/Sms3anc000*,cer/Sns3anc000*,cere/V0p3000iz0*,cere/V
0s1000iz0*,cere/V0s1000cz0*]
locui[locui/V000000f00*,locui/V0s3000is0*]
adipocite[adipocite/null*]
```

In Figure 3 we can see that for some words the program cannot determine the corresponding part of speech, so that the resulting data are then filtered in order to extract the essential information which will be displayed as a table through the application interface. As in the case of the spellchecking process, the results from the grammatical analysis process can also be exported for later access.

3.3. Identifying the discourse type

Identifying the discourse type is one of the key steps of the analysis process. In this phase, in order to streamline the classification process, the introduced or uploaded text is first preprocessed, by eliminating the repeated spaces and tabs and the stop words (for example: “a, dupa, la, cum, astfel, acolo, pana la, despre etc.”). Following, the program runs the actual classification process. Additionally, versatility is obtained by allowing the user to either upload the input file or to directly copy the text in a window of the web page.

Depending on the number of analysed discourse types, the linguistic processing of the texts, used to identify the features for each discourse (like: number of words, frequency of words, total number of words in the text, spelling errors, grammar results, etc.), requires one or more resources, which will form the training data used in the classification process. These training sets are represented then as Java objects, consisting of items such as: the number of words and their frequency of occurrence, the number of types of categories observed and the corresponding class for the analysed text.

DocumentData object is the main entity for data representation. This class stores the extracted information from the input files, which will be used in the training and classification process. This information describes the text properties, such as: all the words from the text, along with their occurrences and the resulting category of the analysed text obtained as a result of the classification process. After the training process is complete, a list consisting of documentData objects will be created, and from the classification process we will obtain the characteristic object that can be classified as one of the discourse type identified in the training phase.

TrainingData object contains the total number of training data, the number of occurrences for each word in a particular category of speech and the number of occurrences of each category in the training data set. All these data, less the total number of training set, are represented as pair of key-value properties.

DataForNaiveBayesClassifier stores all the information obtained from the training process, which will be used by the Naïve Bayes Classifier. This data includes: the number of the types of discourses that can be identified by the classifier, in this case, it will be four, the number of the distinct words from the entire training set, words that will compose the vocabulary of the classifier, respectively, the total number of the words, (not necessarily distinct), followed by the probabilities used in the identification process of the discourse type for each analysed text.

3.4. Training the Naïve Bayes classifier

In order to identify the type of discourse for a certain text, by applying the Multinomial Naïve Bayes algorithm, first it is necessary to form the data that will be used by the classifier. For that, each file from the training set is processed in order to extract the relevant words for the classification process, building the TrainingData object. These data will contain all the information needed for calculating the probabilities used by the algorithm.

After obtaining this information, the program will calculate the probabilities that will be used later, when establishing to what type of discourse, from the training set, the analyzed text belongs to. These probabilities are obtained using the following formulas::

$$P(c) = \frac{Nc}{N} \qquad (1)$$

where Nc represents the number of occurrences of category c in the obtained training set, and N is the number of input data.

$$P(w|c) = \frac{count(w,c) + 1}{count(c) + |V|} \quad (2)$$

where count(w,c) computes how many times the word w appears in class c, plus one (en: "smoothing"), and count(c) represents the total number of words from class c, plus the total number of distinct words from the training vocabulary.

Because it is necessary to process a greater amount of data for the training of the classifier, the training process of the classifier becomes more expensive. In order to avoid that and to reduce the waiting time of processing a classification request and also to improve performance, gathering the data and training the classifier happens only once, at server start-up. This was done by annotating the training method of the classifier with the @PostConstruct8 annotation. This annotation allows the execution of the method to happen immediately after all the initializations were done by dependency injection.

Therefore, all the necessary data for the classification process are calculated only once, and they can be used for a new classification whenever it is necessary, without the need to repeat the training process of the classifier.

3.5. Classification of the text and application of the algorithm

To identify the type of discourse for a certain text by applying the analysis process described earlier, same as in the training process, a DocumentData object is created. This object will contain all the necessary data for calculating the probabilities used for determining the maximum score of the category in which it will be classified.

In case the targeting text contain words that are not present in the training set, they are ignored, because they were not classified as belonging to any category that can be identified by the classifier. Therefore these words cannot influence the result of the classification process.

Further, having the vector with all the words contained by the entered text, the process of determining the type of discourse for the text continues with identifying the category that has the maximum post conditional probability calculated using the formula:

$$P(c|Y) = P(c) + P(y_0|c) * f(y_0) + P(y_1|c) * f(y_1) + \ldots \quad (3)$$

where c is the category of the discourse, $P(c)$ is the probability calculated for the c category, $P(y_i|c)$ is the conditional probability of the word y_i , $f(y_i)$ is the frequency of the word y_i, and Y is the vector of words form the analysed text.

If the type of discourse cannot be identified by the program, the category in which the text will be classified will be: "undefined". This result can be achieved when none of the data from the analysed text can be found in the training set. For example, this may happen when we try to analyse a text that is in another language than Romanian.

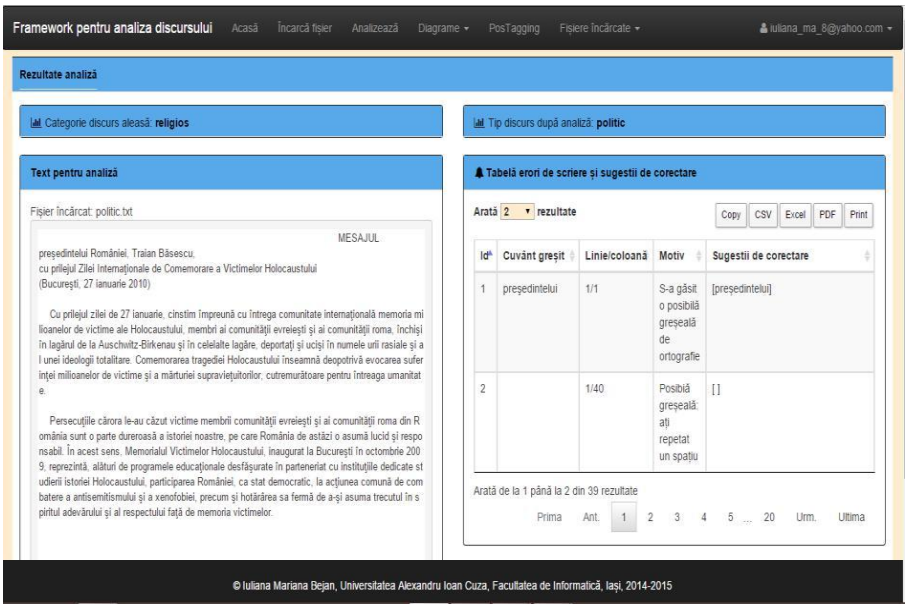


Figure 4. The analysis results

However, because of the fact that identifying the type of discourse is based on the words contained by the text, for the texts in Romanian, the language supported by the application, the probability to achieve that is very small, even negligible. Therefore, even the texts that are not within the scope of the four targeted areas, they will still be classified within the class with the maximum post conditional probability.

Figure 4 shows the results of the analysis performed on a political text. As can be seen the text was correctly classified.

3.6. Diagrams

After the type of discourse was identified by using the Naïve Bayes algorithm applied for the entire file that was uploaded or for the whole typed text, the text is then divided into paragraphs. After that, on the resulted list of paragraphs the classification algorithm is applied again in order to identify the category of each paragraph from the list.

Having these results, the list of ChartDTO's objects are created, which will be used later on for drafting the diagrams and the statistical results of the application. There are 3 types of diagrams supported by the app: *PieChart 3D diagram*, *ColumnChart diagram* or *PolarAreaChart diagram*.

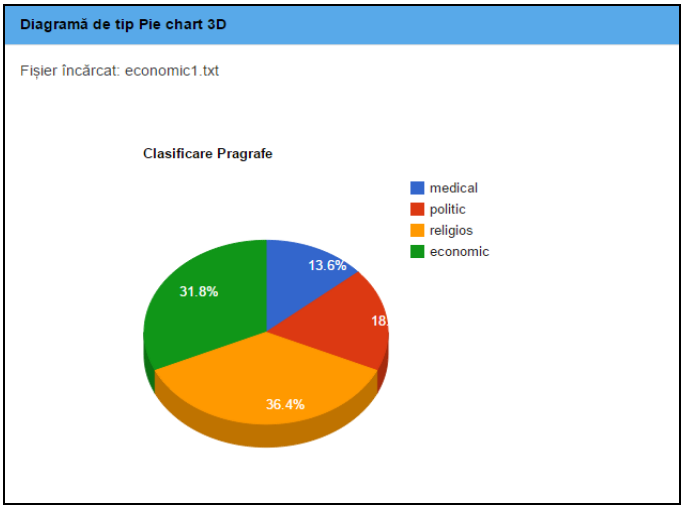


Figure 5. PieChart 3D Diagram of the results

The first two types of diagrams were created by using *Google Charts*[8]API, whereas the third diagram was created by using *Chart.js* API[9]. In each case, the necessary data needed for the diagrams was obtained by sending ajax calls to the server. In Figures 5 and 6 we can see the first two types of diagrams obtained.

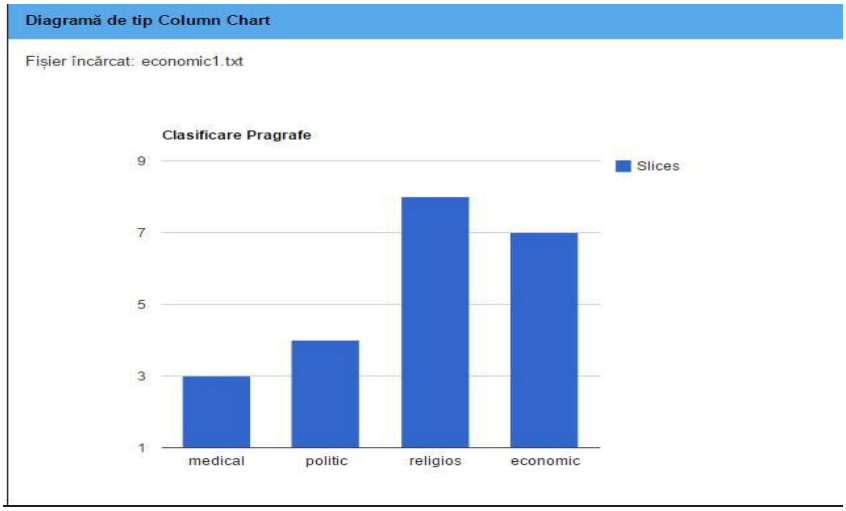


Figure 6 ColumnChart Diagram of the results

[8] Link: https://developers.google.com/chart/interactive/docs/dev/dsl_get_started
[9] Link: http://www.chartjs.org/docs/

The way choosen for showing the diagrams allows the user to easily identify the defining elements used for classification, and the fact that we are using aAjax calls to get that data, speeds up the display in the page of the result, without the need to reload the page. This way, the waiting time spent for accessing the data is also reduced and the user experience within the application, is improved.

Furthermore, having these results, the user can choose to eliminate some words or paragraphs from the text in order to get a better classification of the text, depending on the type of discourse that she/he wants to obtain.

4. Evaluation of the results

4.1 Text/File evaluation

In order to decide the accuracy of the results for a file, the program was used on a set of ten texts: two texts for each category from the area of concern, plus other two for texts from different areas, others then the four mentioned earlier.

Following the analysis carried out of the ten files examined, namely eight of them, corresponding to the four areas of interest, were correctly classified, being associated with the right domain, whereas, the last two, namely those belonging to other areas, were misclassified, being identified as one of the four types supported by the program, although they were from completely different domains.

This happened because those two last texts contained some words from the training set, therefore the program chose the category with the highest probability. In case those texts were made up only of words that weren't present in the training dataset we would obtained as a result of the analyzation process the "undefined" type.

At the same time, we could also notice that the texts that were from other areas than the four supported by the program were usually classified as being from the economical area or from the political area. This is due to the fact that in these two areas, economical and political, we encounter a lot of frequently words that are commonly used in other types of discourses, while, the lexicons of the medical or religious texts, are much more specialized, and much less intersected with those of other domains.

Therefore, for texts that belong to one of the four areas of interest, the accuracy of the classification is between 80% and 90%, while for other texts, the system still tries to fit them into one of the four types, resulting in erroneous classifications.

This limitation in the performance of the application can be overcome, by adding new classes to the training phase.

4.2 Paragraph evaluation

For establishing the classification accuracy for the paragraphs, the program was tested on 100 paragraphs: 20 paragraphs for each type of discourse out of the four mentioned, plus other 20 for other areas.

Out of the 80 paragraphs that fit within the scope of the application, 73 were properly classified in the correct classes, while 5 of the economical paragraphs were misclassified as being from the medical area, and the rest of 2 paragraphs belonging to the economical domain, were identified as being from the political area.

Regarding the remaining 20 paragraphs belonging to other fields, the behaviour of the program was the same as for the files from other fields. Therefore, they were classified in one of the four areas, mostly – as belonging to the political or economical areas. The accuracy of paragraphs classification is pretty good, falling in the range 73% to 80%.

We can see that, for paragraphs, the accuracy is slightly lower than the one for the entire text, and this is because the analysis process takes into consideration fewer words, with a much lower frequency of occurrence.

4.3. Accuracy of classification

It is a known fact that the accuracy of the classification is influenced by the independence of the data in the training dataset. If data are more dependent on each other, then the precision of the classifier decreases. This truism influenced the selection of the four domains that were supposed to be supported by the program: political, economical, religious, and medical.

However, our accuracy was also influenced by the size of the training dataset corresponding to each one of the four domains. As stated earlier, we tried to reduce the impact of a small training set on classification by eliminating stop words and by performing a normalization of the data before executing any other action.

4.4. Duration of the classification process

Since the application was tested for accuracy on a set of 10 texts of various sizes, respectively, 100 paragraphs, we could observe also the workload of the application, the waiting time of a request processing and also the response time. As a result, the processing of a paragraph analyze request was completed in less than 2 seconds, while the text processing request of a complete text, because of the increased size, was achieved on an average from 10, up to 20 seconds. The highest processing time, which was 20 seconds, was registered for the case when the user uploaded a file.
As we can see, both, the waiting time for processing a request and the time for displaying the results, are reduced, and that is because the training of the classifier was executed only once, at server startup, which significantly increased the program's performance.

5. Conclusions and future work

This paper presents how the Multinomial Naïve Bayes algorithm was used in the process of deciding the type of speech for an analyzed text.

Based on the performed analysis, this algorithm has proven to be very effective for

achieving the defined purpose, offering a correct classification in approximatively 90% of the cases. An important aspect of ussing this classifier is that it doesn't consume a lot of the computer resources and the time complexity of the algorithm is linear. Therefore, because the training task is performed only once, at server startup, the text classification process is one of the fastest tasks that run.

Additionally, besides text categorization, the application provides other features including: grammar checking, spellchecking, and the possibility to display statistics with the results of the classification. Also, after the spellchecking process is completed, the user receives a number of suggestions that could be used to correct any error that might have been spotted by the program. By using statistics to display the results, the user experience is improved and the interest in using the application might increase.

In conclusion, the developed application can be used successfully for speech analysis, and it can be very useful when we are interested to perform correct categorization of texts within one of the four areas of interest: politic, economic, religious, medical. Of course, by extending the area of interest of the application, it could be used for other types of speech too, others than the one mentioned.

A first approach for a future work on this topic could be to broaden the areas of interest by adding more fields that can be analyzed and identified. This could be very easily achieved by adding new classes of speech to the training data. This approach would widen the scope of the program, providing viable alternatives for proper drafting of different texts from different fields.

By now the program performs the text analysis at a lexical level, a new future approach would be to identify the semantic level. This approach would improve the reliability of the results and would increase the user interest into using the application. In order to increase the reliability of the results given by the program, the application could be extended by adding a new component responsible for comparing the results obtained by the program himself and the results obtained using other classifier. An example of such situation could be to compare the results of the program with the result obtained using the SVM classifier offered by Weka [10] software.

References

Benveniste, E. (1966). Problèmes de linguistique générale. Paris, Gallimard, pp. 118-131 (republished in 1974).

Frunză O., Inkpen, D., and Nadeau, D. (2005). A text Processing Tool for Romanian Language. In *Proc. of the EuroLAN 2005 Workshop on Cross-Language Knowledge Induction*.

[10] http://www.cs.waikato.ac.nz/ml/weka/

Grivel, L., and Bousquet, O. (2011). A discourse analysis methodology based on semantic principles – an application to brands, journalists and consumers discourses. In Journal of Intelligence Studies in Business 1, pp. 76-86.

Maingueneau, D. (1984). Genèses du discours, Mardaga, Liège, 5.

Pêcheux, M. (1990). L'inquiètude du discours, Editions des Cendres, Paris, 102. Petitjean, A. (1989). Les Typologies textuelles. In *Pratiques*, no. 62, pp. 86-125.

Sources

1. http://www.semantic-knowledge.com/tropes.htm
2. http://gate.ac.uk
3. http://nlp.cs.nyu.edu/oa
4. http://minorthird.sourceforge.net
5. http://www.site.uottawa.ca/~ofrunza/RO-Balie/RO-Balie.html
6. http://balie.sourceforge.net
7. https://docs.oracle.com/javaee/5/api/javax/annotation/PostConstruct.html.

EXPLORING LIST OF MARKERS IN UNSTRUCTURED TEXT AUTOMATIC PROCESSING

MIRCEA PETIC[1,2], SVETLANA COJOCARU[1], VERONICA GÎSCA[1]

[1] *Institute of Mathematics and Computer Science, Moldavian Academy of Sciences, Chișinău*
[2] *Alecu Russo Bălți State University, Bălți*

Abstract

In this study we propose to identify some data source that can be used to determine "controlled vocabulary" (lexicon of markers), in order to develop an engine for extracting articles refering to a given topic. Methods of automatic enriching of the created lexicon of markers are presented.

Keywords: computational linguistic resources, linguistic markers, derivation.

1. Introduction

We live in a dynamic world, characterized by a multitude and variety of events, but also by the promptness of the informational (and not only) reaction to them. One can find a lot of examples illustrating the role of ICT in the organization or remediation of the events with a major social impact: e.g. the so-called "Twitter Revolution" of 2009 in Moldova, which led to early elections, or rapid collection of considerable funds through social networks in helping hurricane victims in Southeast Asia.

Thus, the Internet in general and social networks in particular become an important tool for detecting, monitoring, modelling and mitigating social disasters caused by actions of different nature. In our research, conducted within the NATO Science for Peace Project NUKR.SFPP 984877 – "Modelling and Mitigation of Social Disasters Caused by Catastrophes and Terrorism", we will focus our attention on processing large volume of unstructured data available at global information networks in order to identify and analyse Romanian language texts, which refer to the subject mentioned above.

In this area and other ones related to the researches and applications of natural language processing, we find that most of the works are carried out for English. Since only 26% of Internet users speak English[1], the construction of computational linguistic resources and tools in languages other than English is a growing need. For this reason the European Commission has initiated a number of projects in support of technologization of European languages other than the English language. This policy of promoting multilingualism through information technologies is continued,

[1] www.internetworldstats.com/stats7.html, June, 2015

the fact which can be proved by the priorities set out in the Framework Programme 7 (and continued in current HORIZON 2020 Programme).

The Romanian language becomes one of the significant languages in what concerns informatics resources and technologies applied to them. Therefore, it remains an actual problem to develop new applications and computational linguistic resources for Romanian. In this study we propose to identify some the useful data sources, accumulate and process a limited number of texts in order to create a basic lexicon of relevant words (markers) for disaster domains of different kinds: natural, technological or social. Also, our aim is to develop methods for automatic enriching of this lexicon. Thereafter the formed lexicon of markers will serve as a basis for development of tools to extract texts from the identified sources, to classify them and to provide sentiment analysis.

Our paper is structured as follows: section 2 describes the state of the art in the proposed topic, section 3 presents characteristics of social media in the Republic of Moldova, in section 4 we describe the collection of articles obtained from news sites, sections 5-6 are devoted to the creation of lexicon of markers and its completion by means of derivation.

2. Approaches for processing unstructured texts

Starting with the aims of the projects described above, we will follow the approaches from the research presented in (Banea *et al.*, 2011) concerning multilingual sentiment and subjectivity analysis, but it works also in other cases.

They identified and overviewed three main categories of methods in their research:

1. Those focusing on word and phrase level annotations;
2. Methods targeting the labelling of sentences;
3. Methods for document-level annotations.

In order to make annotation at different levels, they have lexicons with entries learned from corpora. The entries in the lexicons labelled as part of speech for those that appear most often in subjective contexts are strong clues of subjectivity. Those that appear less often, but still more often than the expected by chance, are labelled as weak clues. Each entry is also associated with a polarity label, indicating whether the corresponding word or phrase is positive, negative, or neutral (Banea *et al.*, 2011).

Another lexicon that has been often used in polarity analysis is the General Inquirer (Stone, 1968). It is a dictionary of about 10,000 words grouped into about 180 categories, which have been widely used for content analysis. It includes semantic classes (e.g., animate, human), verb classes (e.g., negatives, becoming verbs), cognitive orientation classes (e.g., causal, knowing, perception), and others. Two of the largest categories in the General Inquirer are the valence classes, which form a lexicon of 1,915 positive words and 2,291 negative words.

SentiWordNet (Essouli and Sebastiani, 2006) is a resource for opinion mining built on top of WordNet, which assigns each synset in WordNet with a score triplet (positive, negative, and objective), indicating the strength of each of these three properties for the words in the synset. The SentiWordNet annotations were automatically generated, starting with a set of manually labelled synsets. Currently, SentiWordNet includes an automatic annotation for all the synsets in WordNet, totalling more than 100,000 words.

Other resources have several corpora. Subjectivity and sentiment annotated corpora are useful not only as a means to train automatic classifiers, but also as resources to extract opinion mining lexicons. For instance, a large number of the entries in the lexicons mentioned above were derived based on a large opinion-annotated corpus. The corpus was collected and annotated from news articles from a variety of news sources manually annotated for opinions and other private states.

There is a large number of approaches that have been developed to date for sentiment and subjectivity analysis in English. The methods can be roughly classified into two categories:

1. Rule-based systems, relying on manually or semi-automatically constructed lexicons;

2. Machine learning classifiers, trained on opinion-annotated corpora.

Among the rule-based systems, one of the most frequently used is OpinionFinder (Wiebe and Riloff, 2005), which automatically annotates the subjectivity of new text based on the presence (or absence) of words or phrases in a large lexicon.

On the other hand, when annotated corpora is available, machine-learning methods are a natural choice for building subjectivity and sentiment classifiers. For example, Wiebe *et al.* (1999) used a data set manually annotated for subjectivity to train a machine learning classifier, which led to significant improvements over the baseline.

The development of resources and tools for sentiment and subjectivity analysis often starts with the construction of a lexicon, consisting of words and phrases annotated for sentiment or subjectivity.

3. Social media in the Republic of Moldova

The analysis of data in order to extract relevant structured information can be performed on different categories of data and for different purposes. In this section, we will mainly refer to analysing data individually published on social sites by their users. Public safety and emergency tools, in the context of the high popularity of social applications and the abundance of individual broadcasting messages issued inside them, must be reviewed and adapted to the new possibilities of possible social threat detection.

The most popular social network in the world is Facebook. Over the past few years the number of registered users has increased in the Republic of Moldova. In March

2015 Facebook reached 500,000 persons[2] in the Republic of Moldova. 65% of these users are between 18-34 years old. 33% of these users set Romanian as their basic language, 9% use English and 56% - Russian. But the favourite resource remains odnoklassniki.ru, which has about 1mln. users in Moldova. There are two strong reasons why Odnoklassniki is in the top in the Republic of Moldova. First, the website is extremely simple to navigate, making it especially attractive for citizens in rural areas who have just gained access to the Internet. Also, the majority of Moldovans speak Russian, which makes the website familiar and closer to Moldovan heritage than other social networks. Moldovans use Odnoklassniki primarily for communication and gaming, and very rarely for promotional campaigns.

One of the commonly used methods for obtaining necessary data samples is via application programming interfaces (APIs) from social media sites. Only a limited amount of data can be obtained daily. Without knowing the populations distribution, it is difficult to extract reliable samples of data. The problem is to find whether the obtained data from social media are some indication of true patterns.

By its nature, social media data can contain a large portion of noisy data. Facebook and other social networks are the place where we can find different promotional campaigns having commercial character, but not only. For this data, we notice that blindly noise removing can worsen the problem stated in the big data paradox because the removal can also eliminate valuable information.

For social networks exists a specific way to attach a message to a selected topic – hashtags. For example, the hashtag `#colectiv` unifies all messages concerning the tragedy in Bucharest. One can observe that the literal sense of hashtags might not refer directly to the disaster topics. For example, `#jesuischarlie` (I am Charlie) contains no words related directly to terrorism, but everyone knows that it is a slogan to express support of freedom of the press after the 7 January 2015 massacre at the French satirical weekly newspaper Charlie Hebdo. Another observation is that messages marked with a hashtag can contain other hashtags. Analysing the previous example regarding to the hashtag `#colectiv` we can find in the messages marked with that hashtag also the marks `#find_us` (corresponding to a photo exposition that will take place in December 2015), `#nuvomuita`, `#bucuresti`, `#tragedie`, `#SRI`, `#Balotești` etc.

Messages marked with these hashtags are characterised by a specific lexicon, with the abundance of words connected with tragedy, victims, infections, hospitals, etc. Another group of hashtags refers to `#refugiat`, `#refugiaților`, with `#refugiati` also presented as well. Together with this hashtag the following markers also are used: `#UE`, `#fonduri`, `#Emotionant`, `#violenta`, `#granita`. Moreover, the hashtags mark not only messages but also articles from the sites.

[2] according to Gramatic agency, http://gramatic.md/blog/500-000-utilizatori-facebook-moldova/

Therefore the process of information collecting from social networks has advantages and disadvantages. On the one hand, we can easily identify a chain of information that relates to a particular topic by following a certain hashtag, on the other – it is difficult to identify automatically the marker itself.

4. Online news articles

The second source of texts was the site www.noi.md from which we collected news articles refering to the topic of social disasters. The collected articles are referred to the topic of social disasters, and classified into three categories: catastrophes, epidemics and disasters.

For the beginning and approving the hypothesis of research we collected 26 articles containing 8450 words. The collection of texts is annotated at sentence and word levels, providing morpho-lexical information using UAIC Romanian Part of Speech Tagger (Simionescu, 2011).

The obtained collection of Romanian texts about disasters permits to extract and annotate a lexicon of markers concerning the topic on disasters. We annotated manually those words and/or word expressions that express the meaning of social disasters. The most frequent words are "catastrofă" (catastrophe), "accident" (accident), "morți" (dead) and "incendiu" (fire). As we established 3 categories of articles of social disasters, each of them has its own lexical markers. Together with the words selected from social networks we obtain 117 markers. 78 words refer to the catastrophes, the epidemics are described by 24 words, and the fire - by 15d.

4.1. Lexicon of markers

As we extracted first the list of markers, we can organize them into a lexicon of markers. Started with those 117 words that express the meaning of disaster, in the future we will complete them with other words from the new texts.

The Republic of Moldova has a Service of Civil Protection and Exceptional Situations which has its own classification of social disasters. It consists of 3 main categories of disasters:

1. Exceptional situations with technical characteristics;
2. Exceptional situations with natural characteristics;
3. Exceptional situations with biological-social characteristics.

Each category has its own subcategories, with notions that describe the social disaster. There are 190 situations described in the official classification. We can see that it is more than we have obtained with our collection, thus we will analyse the similarity and the differences in these two lists of markers and add new texts in order to obtain full coverage for all 190 situations described in the Civil Protection classification.

The third source of lexicon enriching is the controlled vocabulary proposed by H.N.Teodorescu (2015a; 2015b). It contains about 150 words selected from a variety of sources. Also we use the results from (Bolea, 2015) where the words related to two kinds of hazard/emergency situations, earthquake and fire, are analysed and the statistics of the words used in posts on Twitter, Google+ and the website of the Romanian "National Service of Seismic Warning" are presented.

The obtained collection and markers were extracted manually. They serve as a basis for elaboration of an engine which inspects the given list of sites and extracts the articles containing markers. The corresponding software is under development. In its elaboration the date of publishing is taken into account (in order to avoid old posts), a special action is included to delete eventual noise (non-relevant publicity etc.).

4.2. Automatic enrichment of the lexicon of markers

Usually we operate with word-markers in the form of lemma. To extract relevant texts it is necessary to use instruments with rules of inflection and derivation. If the cases of inflection that do not change the meaning the problem is solved automatically (Petic&Cojocaru, 2015), in the process of derivation things are more complicated.

The particularities of the derivational morphology mechanisms help in lexical resources extension without any semantic information. Moreover, there are similar processing mechanisms for different languages spoken in Europe, namely English, French, Spanish, Russian and Romanian. The approaches and mechanisms presented in the paper have been studied on the examples from Romanian, but, in most of the cases, they can be more or less applicable to other languages (Petic *et al.*, 2011). In the following we will examine some of the automatic derivation methods.

5. Substitution of affixes

The idea is inspired by Serbian derivational morphology (Duško and Krstev, 2005), where the generated derivatives have predictable meanings, namely the gender modification in the case of suffix substitution, e.g., *atacator* ↔ *atacatoare* (Eng. *attacker*), and in the case of prefix substitution there is a change of meaning, e.g., *confrunta* ↔ *înfrunta* (*to confront* ↔ *to face*).

In the general case for suffix substitution, let x_1 be a word of the form $x_1 = \omega\alpha_1$ with the suffix α_1. After the substitution $\alpha_1 \rightarrow \alpha_2$ we obtain the word $x_2 = \omega\alpha_2$, e.g., *afectat* → *afecțiune*. In the case of prefix substitution, let x_1 be a word of the form $x_1 = \alpha_1\omega$, where α_1 is a prefix. After the substitution $\alpha_1 \rightarrow \alpha_2$we obtain the word $x_2 = \alpha_2\omega$, where x_2 is the obtained derivative, e. g., *antiseismic* → *preseismic*.

From the information above a new and original algorithm was developed which consists in examining the words in the lexicon and substituting the affixes in those cases that correspond to the categories established by the above-mentioned rules.

6. Formal models

Formal models of derivation rules include a basis, from which derivative words are generated with a high degree of accuracy. A similar approach in derivational morphology is applied for French (Fiammetta and Dal, 2000). But when the French system works with only 3 suffixes (*-able, -ite, -is* (*er*)), for which rules have been found, in the case of the Romanian derivational morphology this study has taken in consideration 3 prefixes (*ne-*, *re-*, *in-*/*im-*) and 2 suffixes (*-re*, *-iza*).

- ***Rules for prefixes:***
  - ✓ *re-* $[\omega]_{inf} \rightarrow [\text{re}\ [\omega]_{inf}]_{inf}$
  - ✓ *ne-* $[\omega'\beta]_{adj} \rightarrow [\text{ne}\ [\omega'\beta]_{adj}]_{adj}$
    $\beta \in \{$*-tor, -bil, -os, -at, -it, -ut,-ind, -înd* $\}$
  - ✓ *in-*/*im-*$=\gamma$ $[\omega'\beta]_{adj} \rightarrow [\gamma[\omega'\beta]_{adj}]_{adj}$
    $\beta \in \{$*-bil, -ent, -ant*$\}$
- ***Rules for suffixes:***
  - ✓ *-re* $[\omega]_{inf} \rightarrow [[\omega]_{inf}\text{re}]_{subst}$
  - ✓ -iza $[\omega'\beta\alpha]_{adj} \rightarrow [[\omega'\beta]_{adj}\ \text{iza}]_{inf}$

6.1. Projection of derivatives

The projection of derivatives represents a method of word formation of the prefixed words from the suffixed words of the same root. According to Spanish researchers, the Spanish verb *amortizar* can be derived with the prefix *des-* obtaining *desamortizar*. Also, *amortizar* can be derived with suffixes *–cion* and *–able*. So, the derivative with prefix *des-* can derive with the suffixes *–cion* and *–able*. The hypothesis is that derivatives can inherit/project the derivatives of the stem with the same suffixes which contributed to the prefixation (Santana *et al.*, 2004). This method can be applied to other languages than Spanish as well; e.g., in English from the root *read* one can form the derivatives *readable* and *unread*, therefore, it is also possible to form the derivative *unreadable*.

Generalising the above notes, we conclude that it is possible to formally present the mechanism for Romanian derivational morphology. Let us consider a Romanian word ω, α - its prefix and β - its suffix. Then, the following relation is valid:

$$(\omega \rightarrow \alpha\omega) \wedge (\omega \rightarrow \omega\beta) \Rightarrow (\omega \rightarrow \alpha\omega\beta),$$

for example, (*a infecta* $\rightarrow$ *a dezinfecta*) $\wedge$ (*a infecta* $\rightarrow$ *infectare*) $\Rightarrow$ (*a infecta* $\rightarrow$ *dezinfectare*) [in Engl.: (*to infect* $\rightarrow$ *to disinfect*) $\wedge$ (*to infect* $\rightarrow$ *infection*) $\Rightarrow$ (*to infect* $\rightarrow$ *disinfection*)];

$$(\omega \rightarrow \alpha\omega) \wedge (\omega \rightarrow \alpha\omega\beta) \Rightarrow (\omega \rightarrow \omega\beta),$$

For example:
(*a capitula* $\rightarrow$ *recapitula*) $\wedge$ (*a capitula* $\rightarrow$ *recapitulaţie*) $\Rightarrow$ (*a capitula* $\rightarrow$ *capitulaţie*) [in Engl. (*to capitulate* $\rightarrow$ *to recapitulate*) $\wedge$ (*to capitulate* $\rightarrow$ *recapitulation*) $\Rightarrow$ (*to capitulate* $\rightarrow$ *capitulation*)];

$$(\omega \rightarrow \alpha\omega\beta) \wedge (\omega \rightarrow \omega\beta) \Rightarrow (\omega \rightarrow \alpha\omega),$$

for example, (*a integra* → *dezintegrare*) ∧ (*a integra* → *integrare*) ⇒ (*a integra* → *dezintegra*) [in Engl. (*to integrate* → *disintegration*) ∧ (*to integrate* → *integration*) ⇒ (*to integrate* → *to disintegrate*)].

Examining the words in the lexicon and verifying them in accordance with the relations above, a new and original algorithm has been developed that generates derivatives by affixes projection.

6.2. Derivational constraints

Where there is no clear model, according to which it would be possible to generate derivatives, some preconditions will appear, called derivational constraints. The most common derivational constraints are: parts of speech, inflection classes, affixes, changes that take place in the case of derivation, the letters preceding/succeeding prefixes/suffixes. So, derivational constraints represent some schemes with several parameters that reduce the class roots and affixes in order to form derivatives.

E.g. functions of the form:

$$f: \{wrd, pos, mod, sla, fgw, mvca\} \rightarrow derivative$$

where *wrd* is a word to derivate, *pos* - part of speech of *wrd*, *mod* – model of derivation, *sla* – the set of letters to which the affix is attached, *fgw* – flection group of *wrd*, *mvca* – modifications and vocalic or consonant alternations (Cojocaru *et al.*, 2009).

Examining the words in the lexicon and verifying them in accordance with the relations above, an algorithm of derivatives generation by derivational constraints has been developed.

As examples of this method of generating derivatives, the automatic derivation of words with the prefix *des-* and suffix *-ime* can be analised.

f: {*a spinteca (to rip)*, verb, *des*<verb>, ...s..., V14, avoid consonant duplication} →*de*(s)*spinteca (to slice)*.

f: {*crud (cruel)*, adjectiv, *des*<adjectiv>, ..., A3, consonant alternation d - z } → *cru*(*d*)*zime (cruelty)*.

Therefore, derivational constraints necessary for the automatic generation process do not depend on just the affix type, but also on the value of the prefix or suffix, considering the fact that each language has its own particularities in the derivation of words.

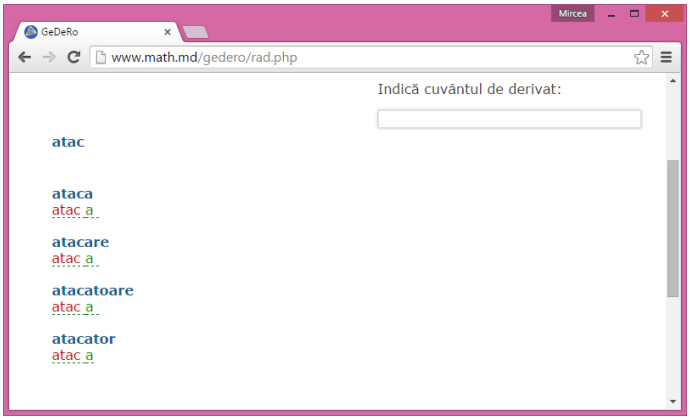


Figure 1. The example for the root *atac*

This public-access Web application (GeDeRo – Generator of Derivatives of Romanian Language) was developed in the framework of independent projects for young researchers and it is supposed to elaborate a set of intelligent control mechanisms which permit interoperable management of available computational linguistic resources.

The following computational linguistic resources that are relevant for our goals were used: DMLR (*Morphologic dictionary for Romanian Language*), RRTLN (*Reusable Resources for Romanian Technology*) and eDCD (*Electronic version of the dictionary of derivate words*).

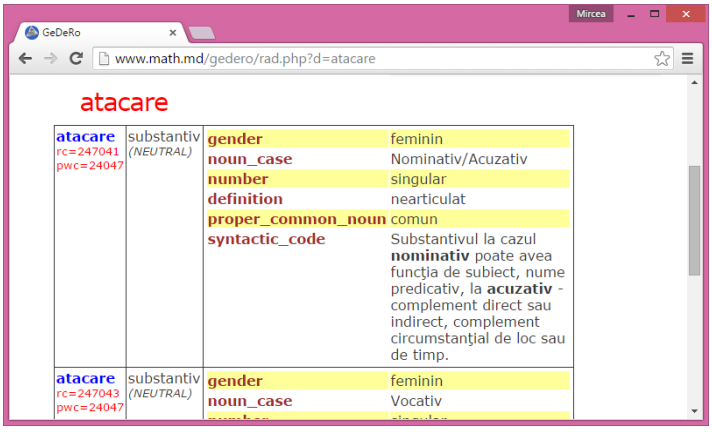


Figure 2. The detailed information about the derivative *atacare*

The starting interface of the public-access Web application GeDeRo[3] offers the possibility to find the derivatives by the prefix, root or the suffix.

Having an input word (for example, *atac – attack*) one can select a criteria stated above and obtain the resulting list of derivatives (Figure 1). This list is connected with RRTLN, where we can see the detailed information about the derived word (Figure 2). When we click on the derivatives, for example *atacare* we will obtain detailed information about this word (Figure 1).

The words obtained after inflecting do not change their meaning, so they will be assigned to the same class of markers, as the lemma-word. In the case of derivation the meaning can vary until the opposite one, and an additional analysis in order to determine the correspondence of derivatives to the initial taxonomy is necessary.

7. Conclusions

Taking into account the list of 117 words obtained from the collection of texts from news sites and social networks, we increase the number of derivate words to 600. Some of them were really very productive in the process of derivation, for example *securitate* (in Engl. *security*) has the following list of derivatives: *electrosecuritate, insecuritate, securist, securistic, securiza, securizant, securizare, securizat, and securizator.* Another word, *a infecta* (in Engl. *to infect*), has also many derivatives, such as: *dezinfecta, dezinfectant, dezinfectare, dezinfectat, infectant, infectare, infectat, nedezinfectat, neinfectat, reinfecta, reinfectare, reinfectat, suprainfectare.* And one more example showing the word productivity in the process of derivation is the word *diagnostic* (in Engl. *diagnosis*) which has the following derivatives: *autodiagnostica, autodiagnosticare, autodiagnostică, diagnostica, diagnosticabil, diagnosticare, diagnosticat, diagnostician, electrodiagnostic.*

383 words out of these 600 derivatives are nouns, 50 – verbs, and 167 – adjectives. The obtained words become later subjects of inflexion. This process generates a number of new different words: 5 for each noun, 10 for adjectives, and 24 – for verbs. Thus the initial 117 words lead to getting a lexicon of 4785 markers.

The obtained lexicon of markers and the proposed methods of its enriching will be used for information processing in order to gather texts related to social disasters. The developed software system has its stand alone value and can be used for a large variety of purposes.

Acknowledgements

The work is carried out as part of two projects: ”Modeling and Mitigation of Social Disasters Caused by Catastrophes and Terrorism" supported by NATO and ”Developing of a text processing system with heterogeneous structure” supported by

[3] http://www.math.md/gedero/

Supreme Council for Science and Technological Development from Republic of Moldova.

References

Banea, C., Mihalcea, R., and Wiebe J. (2011). Multilingual Sentiment and Subjectivity, In: *Multilingual Natural Language Processing*, I. Zitouni and D. Bikel (eds.), Prentice Hall.

Bolea C. (2015). Vocabulary, Synonyms and Sentiments of Hazard-related Posts on Social Networks. An analysis for Romanian messages, in *Proc. IEEE Conf. SPED 2015*, Bucharest, October.

Cojocaru, S., Boian, E., Petic, M. (2009). Stages in automatic derivational morphology processing, in *Knowledge Engineering, Principles and Techniques*, KEPT2009, Selected Papers, Cluj-Napoca, July 2-4, pp. 97-104.

Duško,V., Krstev, C. (2005). Derivational Morphology in a E-Dictionary of Serbian, In Zygmunt Vetulani (ed.), in *Proceedings of the 2nd Language & Technology Conference*, Poznan, Poland, pp. 139-143.

Esuli, A. and Sebastiani, F. (2006). SentiWordNet: A publicly available lexical resource for opinion mining. In *Proceedings of the 5th Conference on Language Resources and Evaluation (LREC 2006)*, Genova.

Fiammetta N., Dal G. (2000). GéDériF: Automatic generation and analysis of morphologically constructed lexical resources. Second International Conference on Language Resources and Evaluation (LREC). Athens, Greece, May 31-June 2, pp. 1447-1454.

H.N. Teodorescu (2015a). Using analytics and social media for monitoring and mitigation of social disasters, in *Procedia Engineering*, vol. 107C, Elsevier, pp. 325-334.

H.N. Teodorescu (2015b) – private mail

Petic, M. and Cojocaru, S. (2015). Vocabulary enriching for text analysis. In Proceedings *of the 17-th System analysis and information technology International conference*, SAIT 2015, Kyiv, Ukraine, June 22–25, 2015, pp. 37-38.

Petic, M. and Osoian, E. (2015). Aspecte de dezvoltare a analizatorului de text nestructurat. In *Proceedings of 5th International Conference Telecommunications, Electronics and Informatics, ICTEI 2015*. May 20-23, 2015, Chișinău, Republic of Moldova, pp. 349-352.

Petic, M., Gisca, V., Palade.O. (2011). Multilingual mechanisms in computational derivational morphology, in *Proceedings of Workshop on Language Resources and Tools with Industrial Applications* LRTIA-2011, Cluj-Napoca, Romania, pp. 29-38.

Santana O., Perez J., Carreras F. and Rodriges G. (2004). Suffixal and Prefixal Morpholexical Relationships of Spanish, Lecture Notes in Artificial Intelligence, Ed. Springer-Verlag, 2004, pp. 407-418.

Simionescu, R (2011). Hybrid POS Tagger. In: *Proceedings of Language Resources and Tools with Industrial Applications Workshop (Eurolan 2011 Summer School)*, Cluj-Napoca, Romania, pp 21-28.

Stone, P. (1968). General Inquirer: Computer Approach to Content Analysis. MIT Press.

Wiebe, J. and O'Hara, T. (1999). Development and use of a gold-standard data set for subjectivityclassifications. In *Proceedings of the 37th annual meeting of the Association for ComputationalLinguistics on Computational Linguistics*, pp. 246–253.

Wiebe, J. and Riloff, E. (2005). Creating subjective and objective sentence classifiers from unannotated texts. In *Proceedings of the 6th International Conference on Intelligent Text Processing and Computational Linguistics* (CICLing-2005) (invited paper), Mexico City, Mexico.

TOWARDS AN AUTOMATIC IDENTIFICATION OF LITERARY AND NON-LITERARY TEXTS

ANDREEA MACOVEI, OANA-MARIA GAGEA, DIANA TRANDABĂȚ

"Alexandru Ioan Cuza" University of Iași, Faculty of Computer Science

{andreea.gagea, oana.gagea, dtrandabat}@info.uaic.ro

Abstract

Classifying text types is an important challenge of the natural language processing field: as there are many texts published and shared every second (literary works, newspapers, blogs, news, laws, etc.), a possible sorting and classification of those texts is extremely necessary. We propose an analysis regarding an automatic classification of several text types: we have implemented a tool that can be used to provide percentages in order to establish if a text belongs to literary or non-literary class. If the determination of text type is made possible, many customized applications can be performed in order to extract information, to analyse the content of the text, to offer suggestions, to summarize a text, to identify a text according to its format, etc.

Keywords: text typology, classification of text types, natural language processing.

1. Introduction

The amount of texts that surrounds us every day needs to be classified before the processing step: this classification is a difficult task as the necessity of a clear and complete text typology is impetuous. In Romanian, this text typology exists for literary works (the classification of literary works by literary genres); in terms of non-literary texts, no classification is clearly established.

Even so, the term of non-literary text is defined in the Romanian literature as a text which aims to objectively inform the reader about certain aspects of reality in a clear and precise language, often containing scientific and technical terms.

This sort of classification needs to be done starting form a series of features that can apply or not for all text types; using these specific features, programs can be performed in order to automatically determine the species, to automatically create content for a certain species, to establish any particular distinction between two types of texts from the same class or not, etc.

This work is part of a theoretical analysis proposed by the authors, regarding the development of a genre ontology: this ontology comprises all the text genres and types specific for Romanian language together with a set of features applied to each species in order to observe the differences and the similarities between various texts.

As the texts are divided into literary and non-literary in the ontology, our main purpose is to emphasize if this classification can be done in an automatic manner: taking into account a number of several textual markers that may prevail or not in a text (cues as *figures of speech, personal or impersonal writing style, presence of neologisms, etc*.), is it possible to automatically delimitate the literary texts from the non-literary ones?

2. Related work

Starting with Aristotle and finishing with the literary theorists of our days, this process of text type classification continues to be the subject of a non-finite taxonomy. Aristotle confirms the existence of several types of poetry and distinguishes between *epic, tragic poetry and dithyrambic poetry* (Simpson, 1988).

On the other hand, in literature, recent studies (Popescu, 2003) in the field of journalism propose a classification of journalistic texts of printed media (*short news, synthesis, editorial, press conference, reportage, interview, etc.*). As these two types of texts are different, it is important to separate them into literary and non-literary classes.

In literature, an increased interest on text classification is visible: studies focus on the structure of a text in order to guess the text type (Pustylnikov, 2007), on the type of writing (and in this case, there are *literary texts, factual texts and persuasive texts*) (Love *et al.*, 2000) and on the representation of texts as a complex network of linked words (Margan *et al.*, 2014).

There are numerous text typologies concerning the scope and sequence of text types proposed by experts (Cooper, 2009), but few of these works concerns the automatic classification. The aspects the most frequent used in order to automatically delimitate different types of texts are content, structure, layout and functions.

Specialists propose structural classifiers of text types (Mehler *et al.*, 2007) that could be an alternative in predicting the function of text (the role of message transmission: *informative role, persuasive role, instructive role, etc*.): this function of text provides significant information that results from that content of text and once this information is extracted, it can be used as pattern in classifying texts.

For our work, we have chosen several characteristics that have can be automatically identified in diverse types of texts. But the question remains: is it possible to determine the class to which a text belongs, using just statistics over a set of features?

3. Corpus

The corpus established for this analysis comprises texts from both categories (literary and non-literary texts; we have chosen to test our tool on 10 texts of each species). For all the species considered in the genre ontology, relevant examples have been found: several texts (especially *informative and lyric texts*) have been

extracted from CoRoLa (Corpus for Romania Language) (Barbu *et al.*, 2014), while the rest of examples from *newspapers, books, contracts, brochures, etc*. In Table 1, all the species of both literary and non-literary classes used for the application are exposed; this classification is proposed by the authors after consulting various studies concerning the theory of literature.

Table 1. Species of literary and non-literary texts

Class	Sub-class	Species
Literary texts	Epic	Diary, Biography, Autobiography, Memories, Epic, Novel, Sketch story, Short story, Novella, Fairytale, Myth, Parable, Ballad, Fable, Poem, Anecdote;
	Lyric	Elegy, Ode, Pastel, Meditative poetry, Satire, Pamphlet, Sonnet, Rondo, Ghazel, Gloss, Romance, Hymn, Doina, Idyll, Haiku;
	Drama	Drama, Comedy, Tragicomedy, Tragedy, Theatre of the absurd;
Non-literary texts	Argumentative	Argumentative texts;
	Informative	Weather reports, Reportage, Press article, News, Reviews, (Instruction) Manual, Scientific texts;
	Instructive	Manuals, Recipes, How-to guides texts;
	Persuasive	Offers, Advertisements texts;
	Descriptive	Travel guide, Descriptive texts;
	Juridical	Contracts and dispositions, Law texts, Regulations.

4. Features

A number of six features have been established as indicators of texts appurtenance to literary and non-literary classes. Their list is provided below:

- personal or impersonal indications that can be visible in texts through verbs, pronouns, pronominal adjectives (1st, 2nd and 3rd person singular and plural), etc.
- subjective aspect of a text provided by imperative mood of verbs and vocative case of nouns;
- length of a text;
- presence of figures of speech (comparisons, enumerations and repetitions);

- marks of temporality and spatiality dimensions (temporal and spatial adverbs);
- frequency of words as a characteristic that may be useful in determining the lack or the existence of neologisms in texts.

Once establishing the characteristics, the challenge is to prove that these general features can serve in the automatic identification process of literary and non-literary texts. In terms of personal or impersonal indications (marks of direct or indirect involvement of the author), things are relatively simple: 1st and 2nd person indications are specific to literary works, while 3rd person indications to non-literary texts. But, as usually, there are exceptions: for example, realistic novels abound in 3rd person pronouns or verbs.

Here are two examples (a fragment of a diary and of a scientific text) where personal or impersonal indications are highlighted. Verbs such as *poți*, *ești*, *dai* indicate the personal character of the text of Example 1, while the impersonal character appears in Example 2 (*există*, *asumă*, *sunt*).

Example 1: Literary text

La o anumită vârstă începi să-ți dai seama cât eşti de singur în lume ca om sau ca individ. În realitate nu există nici rude, nici prieteni cu care să poți fi într-o adevărată şi desăvârşită comuniune sufletească.

[You begin to realize at a certain age how lonely you are in the world as a human being or as an individual. In reality there are no relatives or friends with whom you can be in a real and perfect communion of mind.]

Example 2: Non-literary text

Deşi teoria downsiană asumă existența unei singure dimensiuni ideologice relevante, de cele mai multe ori în viața reală există mai mult de o dimensiune relevantă, motiv pentru care modele ulterioare propun stabilirea poziției partidelor şi a votanților într-un spațiu multidimensional (Davis şi Hinich, 1968; Hinich şi Pollard, 1981; Enelow şi Hinich, 1984), determinat de direcțiile de politici care sunt propuse în campania electorală.

[Even if Down's theory demonstrates the existence of a single relevant ideological dimension, there is more than one relevant dimension in most real-life situations, which is why further models suggest to establish the position of parties and voters in a multidimensional space (Davis and Hinich, 1968; Hinich and Pollard, 1981; Enelow and Hinich, 1984) determined by the policy directions that are proposed in the campaign.]

The imperative mood of verbs and vocative case of nouns is also a feature specific that may appear in literary works. In non-literary texts, such marks have a lower frequency (only verbs in imperative mood), especially in regulations, instruction manuals and advertisements.

For Example 3, the literary text is represented by several verses of a fable where the noun potaie is a noun in vocative case, while in Example 4, verbs in imperative mood such as nu instalați and nu deteriorați indicate the fragment of text is extracted from a user manual.

Example 3: Literary text

- *"Noi, fraţii tăi? răspunse Samson plin de mânie,*

Noi, fraţii tăi, potaie!

O să-ţi dăm o bătaie

Care s-o pomeneşti.

[We, your bothers? furiously replied Samson/ We, your borthers, pooch!/ We'll beat the pants off you/ And you will remember it.]

Example 4: Non-literary text

Înainte de instalare verificați avariile externe. Dacă ele există, nu instalați aparatul.

Nu instalați sau depozitați mașina unde poate fi expusă intemperiilor vremii.

Nu deteriorați butoanele de comandă.

[Before installation check external damages. If they appear, do not install the device. Do not install or store the machine in a place where it may be exposed to weather. Do not damage the controls.]

The length of a text can be a generic feature of texts: often, literary texts (except several species of lyric sub-class as pastel, sonnet, rondo, haiku etc.) are larger than non-literary texts as news, weather reports, recipes, offers etc.

Texts as laws, manuals and brochures are an exception to the rule. But, if we add at this feature several restrictions (the format of a text could be one of restrictions), the length of texts can be emphasized as a feature that delimitates the non-literary species from literary species (for example, laws have a certain format given by the abbreviations such as art. which can be easily identified).

If a text is classified as literary, the figures of speech, the marks of lexical creativity (pejorative verbs, superlative adjectives, nouns with positive and negative connotations) and the presence of interrogations are clearly highlighted in that text.

However, in offers or advertisements, even news (Example 5 exposes a suggestive news title), the author wants to draw the attention of a reader and uses rhetorical questions (those questions are often considered a figure of speech); even if, the place of those questions should not be in an informative text (the informative text must provide information to the reader about something in an objective manner), this rule is not often observed.

Exemple 5: Non-literary text

Iohannis l-a primit la Cotroceni pe ambasadorul american Hans Klemm. Despre ce au discutat?

[Iohannis received the US Ambassador to Romania, Hans Klemm at Cotroceni. What did they discuss?]

Another feature treated in this analysis concerns the temporality and spatiality aspects of a text. As expected, texts such as weather reports, travel guides and literary works contain temporal and spatial indications: our program is tested on small texts and the probability of finding many temporal and spatial indications in non-literary (Example 6) texts is higher than in literary texts (Example 7).

Example 6: Non-literary text (weather report)

Meteorologii au emis luni dimineață o informare de ploi torențiale, care vor afecta aproape întreaga țară, în intervalul 19 octombrie, ora 15 – 21 octombrie, ora 15.

[On Monday morning, forecasters issued a briefing regarding torrential rains affecting almost the entire country, between October 19, 15 pm to October 21, 15 pm.]

Example 7: Literary text (autobiography)

La Liceul Spiru Haret am fost singurul dintre patru elevi israeliți care nu am venit cu certificat de la rabin, ci am invățat religia creștină, avându-l drept dascăl pe preotul Gheorghe Georgescu, om de ispravă, care mă simpatiza și-mi da note mari. Bacalaureatul l-am luat în 1929, urmând apoi cursurile Facultății de Drept și Litere.

[At Spiru Haret college, I was the only one of four students Israelites who did not come with a certificate from the rabbi, but I learned about the Christian religion, with the priest Gheorghe Georgescu as a teacher, a great man who sympathized me and gave me high scores. I took the baccalaureate in 1929 and decided to attend the courses at the Faculty of Law and Letters.]

The lexicon or the vocabulary used by an author can represent a key feature helpful in classifying texts. As literary works (Example 8) are characterized by a common vocabulary, neologisms and specialized words occur very frequently in non-literary texts (Example 9).

Example 8: Literary text (autobiography)

Sunt născut în anul 1912, într-o margine de București, unde tatăl meu, inginer, conducea o fabrică de mobile și cherestea (comuna sub-urbană Pantelimon). Din copilărie m-au atras clopotele și obiceiurile creștinești.

[I was born in 1912 on the outskirts of Bucharest, where my father, an engineer, was leading a furniture and timber factory (the sub-urban village of Pantelimon). Since childhood the bells and Christian traditions attracted me.]

Example 9: Non-literary text (scientific text)

Astfel, un partid poate avea poziţii diferite pe diferite dimensiuni. La fel şi votanţii, motiv pentru care opţiunea de a vota pentru un partid sau altul se face în funcţie de media ponderată a pătratului distanţei euclidiene dintre poziţia partidului şi poziţia votantului pe fiecare dintre dimensiuni.

[Thus, a party can take different positions in different dimensions. It is the case of voters and this is why the option to vote for one party or another is based on the weighted average of the squared Euclidean distance between the party and voter positions on both dimensions.]

5. Identification of features

Once the corpus collected (fragments of both literary and non-literary texts), each input text is processed using POS-Tagger for Romanian Language before starting the automatic identification of structures.

The information provided by POS-Tagger, the built patterns and the mathematic calculations are used in order:

- to extract verbs, pronouns, pronominal adjectives of 1st, 2nd and 3rd;
- to identify figures of speech such as enumerations, repetitions and comparisons;

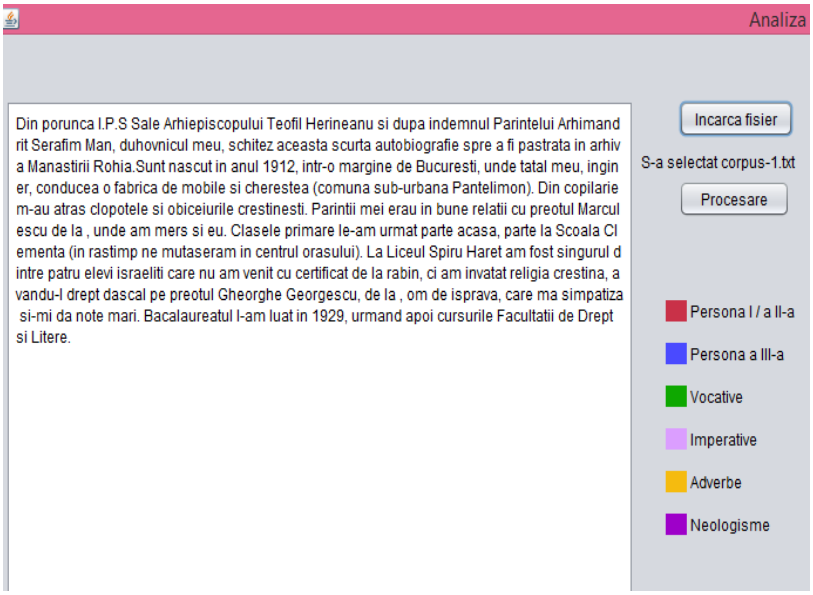


Figure 1. Input sequence of application

- to extract temporal and spatial adverbs;
- to emphasize the rhetorical marks as nouns in vocative and verbs in imperative mood;
- to establish the frequency of words and the length of each text.

For each text, the tool shows the exact number of verbs, pronouns, pronominal adjectives, the adverbs, the nouns in vocative and the verbs in imperative. Using a dictionary of neologisms, it is possible to observe if the text contains or not some specialized words. Figure 1 shows the way how the user can introduce a text and on Figure 2, all the features found in text are highlighted.

6. Text type classification

As mentioned above, the established features appear in all the texts (in a higher or lower proportion). Aiming to classify the texts using these features, for each text, a proportion is fixed according to the numbers of identified structures (as it is shown at the bottom of Figure 2).

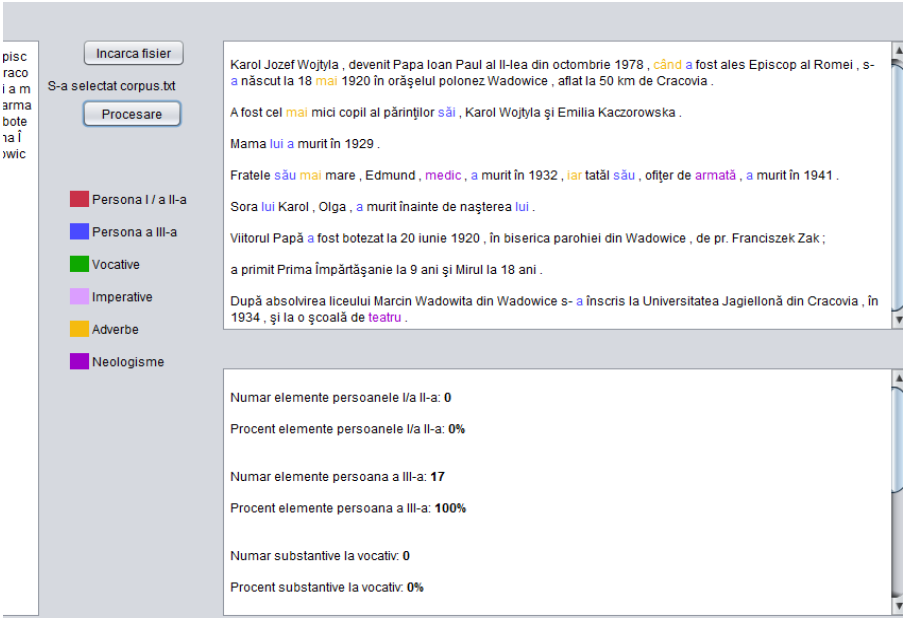


Figure 2. Output sequence of application

After identifying those markers, percentages are calculated in order to establish their frequency in texts. According to these percentages, we can roughly say if a text is literary or non-literary

For the first example, there are approximately 60% chances that the proposed text is literary; for the next example, the application indicates 80% non-literary text.

Example 10: Literary text (diary)

Slăbiciunea şi nevoile vieţii ***te*** *fac să-****ţi*** *închipui că* ***înţelegi*** *pe alţii, că* ***iubeşti*** *şi* ***eşti*** *iubit şi înţeles. Eroare grea, despre a cărei realitate dureroasă* ***ajungi*** *să-****ţi dai seama*** *tocmai în clipele când singurătatea, majestoasă şi divină,* ***te*** *copleşeşte mai crâncen.*

[The weakness and needs of life make ***you*** *consider* ***you*** *understand others,* ***you*** *love and are loved and understood. This is a great mistake about whose painful reality* ***you*** *get to precisely realize in moments when the majestic and divine loneliness overwhelms* ***you*** *terribly.]*

Example 11: Non-literary text (company brochure)

Marc creativ consulting este o companie tânără, ce are ca obiect principal de activitate creația și furnizarea serviciilor de publicitate dispunând de o echipa de specialiști în publicitate, susținută de echipamente tehnice de ultimă generație.

[Marc creativ consulting is a young company, whose main activity is the creation and delivery of advertising services, featuring a team of advertising specialists, supported by cutting-edge technical equipment.]

7. Problems encountered

The problems that we have encountered while testing the application refer to some errors that appear while the text is processed with POS-Tagger and the common features for both literary and non-literary class. No all the verbs in imperative mood of a text are identified using POS-Tagger (verbs as *faceți*, *luați*, *prinde*); it is also the case for rhetorical terms as interjections.
Another aspect concerns the fact that the frequency of words does not provide a clear overview of a text, especially if that text is a small one.
Unfortunately, temporal and spatial indications predominate in both literary and non-literary texts; in this case, another feature is necessary in order to easily classify the texts: format of a text, frequency of figures of speech, direct or indirect involvement of the author, etc.

8. Conclusions

This is a work in progress: using the identified features and the performed tests, some rules will be established in order to assure an accurate identification of literary and non-literary text.

This proposed classification by text type can reveal a series of characteristics of the content of a text. Features as length, spatial and temporal dimensions, figures of speech, direct and indirect marks of the author, etc. can illustrate the diversity of

texts. At the same time, it could be a premise in detecting the style of each author, the creativity of his text and the level of involvement in order to draw the attention of a specific public.

Our analysis demonstrates that depending on the presence of those common features, a text can be classified as literary and non-literary; also, the exceptions that can appear show that the role of an non-literary text has changes as the author does not want anymore to inform the reader, but also to impress and to attract a wide audience by his writing.

Acknowledgements

This work was co-funded by the European Social Fund through Sectorial Operational Programme Human Resources Development 2007 – 2013, project number POSDRU/187/1.5/S/155397, project title "Towards a New Generation of Elite Researchers through Doctoral Scholarships."

References

Simpson, Peter. (1988). Aristotle on Poetry and Imitation. Hermes Press Publishers, pp. 279-291.

Popescu, Cristian Florin *(2003).* Manual de jurnalism. Redactarea textului jurnalistic. Genurile redacţionale, Editura Tritonic, Bucureşti.

Pustylnikov, Olga (2007). Guessing Text Type by Structure. In *Proceedings of the ESSLLI Student Session'07*, pp. 221-231.

Love, K., Pigdon, K., Baker, G., with J. Hamston. (2000). BUILT: Building understandings in literacy and teaching, 2nd Edition Information Division, University of Melbourne.

Margan, D., Meštrovic, A., Ivašic-Kos, M., & Martincic-Ipšic, S. (2014). Toward a Complex Networks Approach on Text Type Classification. In *International Conference on Information Technologies and Information Society* (ITIS2014), November.

Cooper, J. David (2001). Using different types of texts for effective reading instruction. In Current Research in reading / language arts, Boston: Houghton Mifflin, retrieved from http://www.eduplace.com/state/author/jdcooper.pdf in November 2015.

Mehler, A., Geibel, P., & Pustylnikov, O. (2007). Structural Classifiers of Text Types: Towards a Novel Model of Text Representation. In *LDV Forum*, vol. 22, no. 2, pp. 51-66.

Barbu Mititelu, V., Irimia, E., Tufiş, D. (2014). CoRoLa – The Reference Corpus of Contemporary Romanian Language. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation – LREC-2014*, pp. 1235-1239.

CHAPTER 5
LANGUAGE PROCESSING TOOLS

EVALUATING THE COMPLEXITY OF ONLINE ROMANIAN PRESS

MIHAI DASCĂLU[1], DANIELA GÎFU[2]

[1] *Computer Science Department, University Polytechnica of Bucharest*
[2] *Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iași*

mihai.dascalu@cs.pub.ro, daniela.gifu@info.iasi.ro

Abstract

In order to perform a side by side analysis of articles signed by different journalists, we introduce a pilot study performed on the daily *Adevărul* using the ReaderBench multilingual assessment framework. The aim of this study is to evaluate different features of texts reflected in textual complexity indices adapted for Romanian language. The conducted experiments cover a wide range of factors from surface indices to semantics, the later being centred on cohesion. Moreover, of particular interest was to identify semantic similarity links between articles and to cluster them automatically based on the writing styles similarities. Our study was targeted to support journalists, readers, as well as specialists and researchers in the field of natural language processing, linguists and psychologists by providing a wide range of complexity indices that can be used to predict reading ease, adaptability to targeted audience and learner comprehension.

Keywords: textual complexity, online articles assessment, cohesion-based discourse analysis.

1. Introduction

The idea of analysing texts' complexity has long been an important, but difficult task of communication because the measure itself is relative to the reader's understanding level. In other words, the perception of difficulty for a given reading material may be altered due to different reasons such as: previous knowledge in that area, familiarity with the language, personal motivation or interest for the subject presented.

How can we best aligning reading materials to the level of readers?

Text complexity can be defined as the level of reading and understanding difficulty for a text based a series of factors: the readability of the text, the potential levels of meaning, contests or purposes derived from the text, its structure, the conventionality and clarity of the language, as well as specific knowledge requirements. For instance, workplace reading, measured in Lexiles, exceeds grade 12 complexities significantly, although there is considerable variation (Stenner *et al.*, 2010).

As specific goal for our study, we are interested in the vocabulary and discourse difficulty of newspapers, which remained stable over the 1963–1991 period that Hayes and his colleagues (Hayes *et al.*, 1996) studied. We have proposed and assessed a multi-dimensional analysis (Dascălu *et al.*, 2015) of textual complexity initially developed for English and French languages, covering a multitude of indexes integrating surface indexes derived from automatic essay grading techniques, syntax as well as semantics indexes (Dascălu *et al.*, 2012).

The motivation for this topic is to clarify and describe journalistic profiles (Gîfu and Cristea, 2012; 2013) in order to understand anonymous reader's reactions (Gîfu and Cioca, 2013; Gîfu *et al.*, 2013) by comparing textual complexity indices on different types of online contributions (articles versus comments). These textual features are influenced by the amount of available media texts, regardless of their nature and purpose. Therefore, our broader goals presume a better understanding of the media consumer behaviour.

The paper is structured into five sections. After a brief introduction about the importance of this study, section two mentions relevant works focused on textual complexity. The third section briefly describes the analysis indexes adapted for Romanian language, while the fourth section presents a case study on Romanian print press and the statistical results. The last section highlights conclusions and mentions for the future work in order to improve the accuracy of obtained data.

2. Previous work

In our specific analysis context, textual complexity refers to how journalists express personal opinions about different public topics at an appropriate level of rigor and expressivity. In general, online newspapers have forums, where anonymous readers leave comments in different styles. Our interest is to measure the difficulty of online materials and to classify them based on the similarities of writing styles. This will enable us to classify readers based on the previous criteria, as the lecture of online articles by various users is in most cases superficial.

Textual complexity is linked to cohesion in terms of comprehension (McNamara *et al.*, 2012). An anonymous reader must first create a well-connected representation of the article withheld, a situation model (van Dijk and Kintsch, 1983). This connected representation is based on the linkage of related textual pieces of information that occur throughout the article.

Multiple automated systems were developed in order to evaluate textual complexity. For instance, E-Rater (Powers *et al.*, 2001) automatically measures essay complexity by extracting a set of features representing facets of writing quality (discourse structure, syntactic structure, topical analysis). E-Rater supports a multi layered textual complexity evaluation based on the centering theory about building a model for assessing the complexity of inferences within the discourse (Grosz *et al.*, 1995). In 2001, Powers and his colleagues (2001) considered a wider set of indices to measure complexity such as: spelling errors, content analysis based on vocabulary

measures, lexical complexity/diction, proportion of grammar and of style comments, organization, and development scores and features rewarding idiomatic phraseology.

In general, various text analysis platforms were adopted as educational systems (Nelson, Perfetti, Liben, and Liben, 2012), out of which the most representative are: Lexile (MetaMetrics), ATOS (Renaissance Learning), Degrees of Reading Power: DRP Analyzer (Questar Assessment, Inc.), REAP (Carnegie Mellon University), SourceRater (Educational Testing Service) and Coh-Metrix (University of Memphis) and Dmesure (Université Catholique de Louvain). Our experiments are based on the *ReaderBench* platform (Dascălu, 2014; Dascălu *et al.*, 2014) that integrates the most common indices from the previous systems as baseline, but is centred on semantics and discourse analysis. Additional indices for evaluating textual cohesion and discourse connectivity are considered, enabling a more in-depth understanding of the discourse structure.

3. Textual complexity indices adapted for Romanian language

3.1. Surface analysis

Surface analysis addresses lexical and syntactic levels and consists of measures computed in order to determine indexes like fluency, diction or basic statistical analyses by taking into account lexical and syntactic elements (e.g., words, commas, phrase length, sentence and paragraph structure).

3.2. Trins and proxes

Page (1966; 1968) considered that computers can be used, as effective as human teachers, to automatically evaluate and grade student essays by applying only statistical and easily detectable attributes, (Wresch, 1993). In order to perform a statistical analysis of online press that better quantifies the complexity of a journalistic article, we have customized Page's concepts of *proxes* (computer approximations of interest) with human *trins* (intrinsic variables – human measures used for evaluation). As initial studies reported a strong correlation (.71) similar to the inter-human correlations, and proved that computer programs could predict grades quite reliably, our method goes beyond Page's introduced proxes. First, we have considered Slotnick's method (Slotnick, 1972; Wresch, 1993) of grouping proxes based on their intrinsic values, out of which multiple categories were integrated within our model. Second, more in-depth indexes were developed, presented in detail later on.

3.3. Entropy

Entropy, derived from Information Theory (Shannon, 1948; 1951), provides relevant insight regarding textual complexity at character and at word level by ensuring diversity among the elements of the analysis. In other words, a more complex text contains more information and requires more memory and more time for the reader

to process. Entropy means disorder, reflected in the diversity of characters and of word stems used.

3.4. Individual word complexity indexes

Word complexity was treated as a combination of the following factors: distance between the inflected form, lemma and stem, inverse document frequency from the training corpora, the distance in hypernym tree from the WordNet in Romanian[1] (Tufiș *et al.*, 2008), as well as the word polysemy count from the previous lexicalized ontology. While addressing the differences between the inflected form, the lemma and the stem of a word, it becomes clear that a correlation exists between the complexity of a word's derivation and its overall complexity – as multiple prefixes and suffixes are juxtaposed, the more complex the word can be considered.

The distance within the hypernym tree to the ontology root can be seen as a measure of word specialization and specificity. In other words, the more elaborated the path to the root of the ontology hierarchy, the more specific the text can be considered as it introduces more peculiar terms. The closer we are to the root, the more general a text can be considered as broader concepts are used.

In terms of the mean polysemy count per word, we operate under the assumption that the more possible senses a word has, the more difficult it would be to use in a text and to correctly identify its underlying sense. Therefore, simpler texts will contain words that are less ambiguous, while more complex texts, on the whole, will use more words with a higher sense count. The complexity of the constituent analysis elements (phrases, paragraphs, entire document) is obtained as the average of individual scores of content words. A content word is defined as a lemmatized dictionary listed concept after the initial NLP pipeline processing has been applied (Manning and Schütze, 1999), not included in the stop-words list, and having as part of speech: noun, verb, adverb or adjective.

3.5. Semantics

Textual complexity is linked to cohesion in terms of comprehension (McNamara *et al.*, 2012) as cohesion reflects the links between related pieces of textual information that occur throughout the text. Therefore, cohesion reflected in the strength of inner- and inter-paragraph links extracted from the cohesion graph influences readability, as semantic similarities govern the understanding of a text. In this context, sematic cohesion is evaluated at a macroscopic level as the average value of all links added within our cohesion graph (Trausan-Matu, Dascălu and Dessus, 2012; Dascălu *et al.*, in press) (see Figure 1), our core representation of discourse structure.

[1] http://www.racai.ro/tools/text/rowordnet/

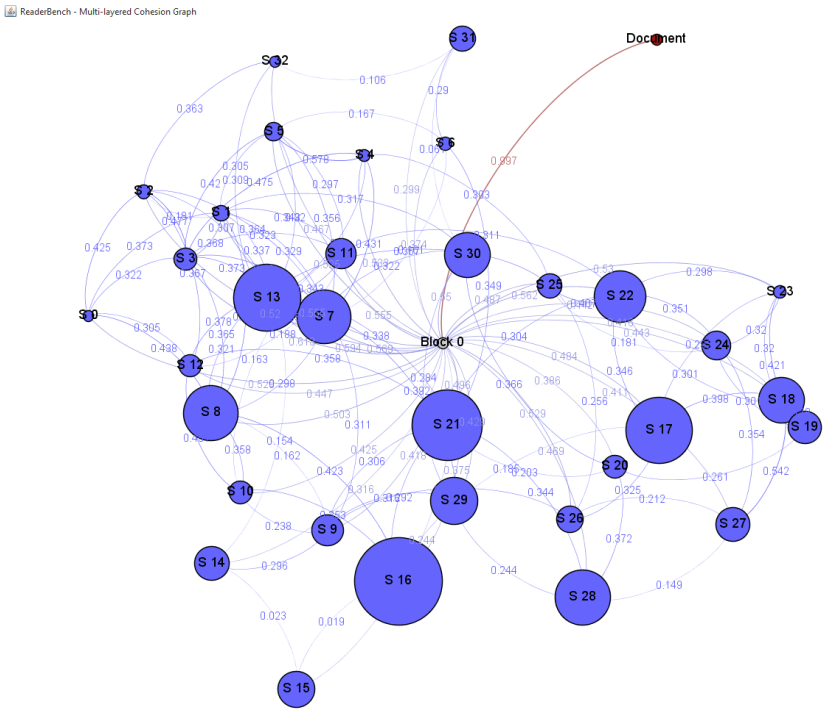


Figure 1. Sample of multi-layered cohesion graph

4. Case Study

4.1. Corpus selection

Our case study is focused on a communication crisis between the Prime Minister and President about the Silvan Code (14 May 2015) to reflect in textual complexity indices, as well as cohesion between articles. All articles on the selected topic (Silvan Code) from the newspaper *Adevărul* have been monitored, stored and pre-processed (tokenization, stemming and lemmatization) in the period May 13-15 2015, structured as follows: May 13 – 2 articles (1248 words); May 14 - 11 article (6451 words); 15 May - 3 article (859 words).

4.2. Training semantic models

Our experiment was preceded by the laborious task of building a corpus of more than 2 million content words, a structured collection of contemporary Romanian texts that covers a wide range of linguistic registers: journalistic, literature, science, religion, etc. and social origins (bookish language worship, suburban language or

slang, and so on). This specific corpus was necessary for training data driven LSA[2] (*Latent Semantic Analysis*) and LDA[3] (*Latent Dirichlet Allocation*) semantic models. The LSA vector space had 300 dimensions, whereas the optimal number of topics for the LDA model was inferred via Hierarchical Dirichlet Processes (Teh *et al.*, 2006) and was set to 175.

4.3. Method and results

In order to perform a thorough analysis of text complexity, several steps were performed.

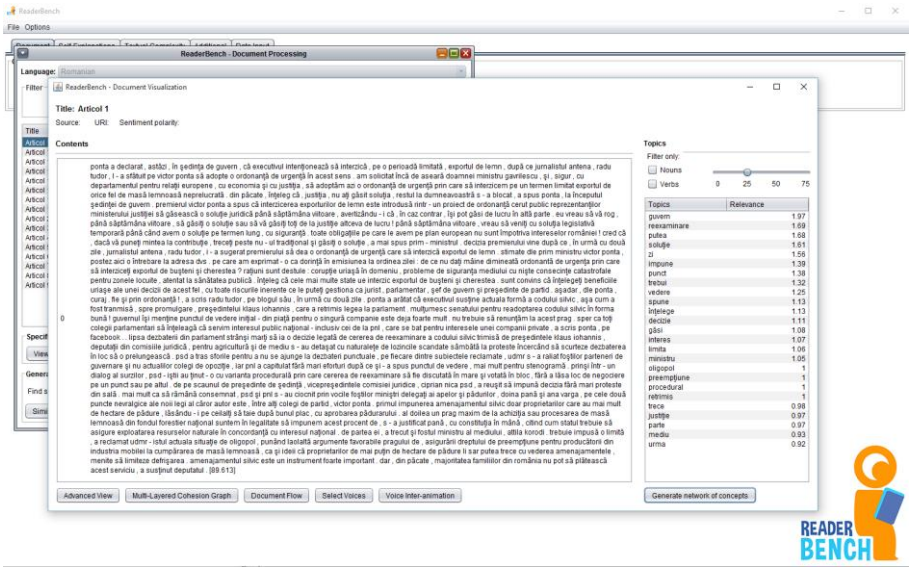


Figure 2. *ReaderBench* main view

Step 1. Apply initial NLP processing pipeline, transpose documents into LSA and LDA semantic models, and build the cohesion graph as central discourse structure in order to reflect cohesive links between analysis elements (document > block / paragraph > sentences) (Figure 1).

Step 2. After the pre-processing phase, use *ReaderBench* multilingual platform to determine the keywords of each article in order to better grasp its specificity (Figure 2).

[2] LSA (Landauer and Dumais, 1997) is a theory and method based on a vector space that is used for extracting and representing the contextual-usage meaning of words by statistical computations applied to a large corpus of text – singular value decomposition on the term-document matrix followed by a projection on *k* dimensions of representation.

[3] LDA (Blei *et al.*, 2003) is a generative probabilistic model of topic modeling in which words and documents are represented as probability distributions in topic classes.

Step 3. Determine specific complexity indices presented in the third section for all articles in the dataset (Table 1).

Table 1. Textual complexity index values for considered articles.

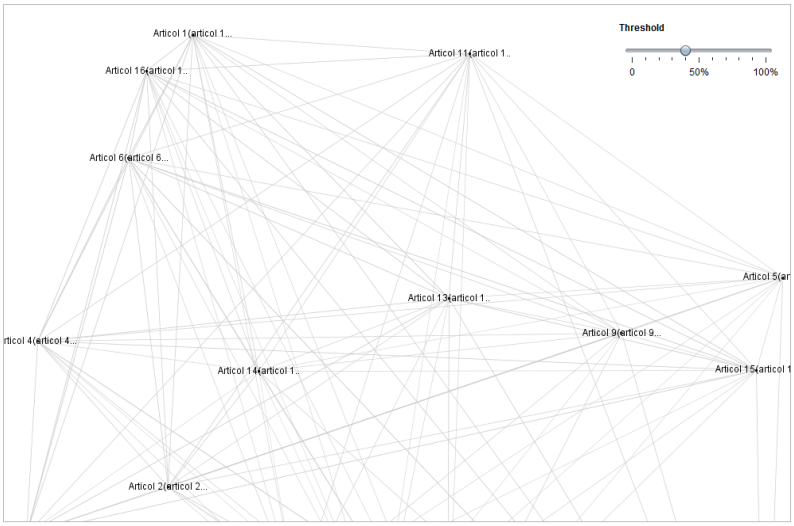
ID / Complexity Index	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Surface indices																
Average paragraph length (characters)	4347	2729	1070	1392	2513	2870	893	2245	1118	1039	1263	2249	3188	958	1249	1491
Average sentence length (characters)	132	210	178	146	109	125	149	94	140	109	230	150	116	113	139	157
Average word length (characters)	5.11	5.91	5.23	5.36	5.07	5.19	5.46	5.15	5.3	5.14	5.55	5.18	5.08	4.93	4.99	5.18
Standard deviation for words (characters)	3.03	3.4	3.26	3.41	3.04	3.24	3.19	3.08	3.09	3.01	3.47	3.18	3.13	2.98	3.13	3.22
Average number of commas per paragraph	82	26	16	18.5	46	41.5	22	23.5	18	24.5	15	40	54.5	10.5	23.5	23
Average number of commas per sentence	2.48	2	2.67	1.95	2	1.80	3.67	0.98	2.25	2.58	2.73	2.67	1.98	1.24	2.61	2.42
Average number of sentences per paragraph	33	13	6	9.5	23	23	6	24	8	9.5	5.5	15	27.5	8.5	9	9.5
Paragraph standard deviation in terms of no. Sentences	0	0	4	8.5	0	21	4	22	5	8.5	4.5	13	25.5	6.5	8	7.5
Average number of words per paragraph	851	462	205	260	496	553	164	436	211	202	228	434	628	195	255	288
Paragraph standard deviation in terms of no. Words	0	0	139	224	0	485	114	393	165	150	175	377	586	156	220	231
Average number of words in sentence	25.79	36	34	27	22	24	27	18	26	21	41	29	22.82	23	28	30
Sentence standard deviation in terms of no. Words	15	13	18.11	18	16	16.26	10.07	10	23	14	22	19	12.74	15.72	14.43	14.58
Average number of unique content words per paragraph	265	163	78	92	183	156	59	95	55	67	86	121	177	65	78	93

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Paragraph standard deviation in terms of no. unique content words	0	0	50	73	0	129	40	73	34	44	64	96	158	47	63	67
Average number of unique content words in sentence	12	19	15	12.42	10	10.22	12.58	7	8	8.37	18.64	13	9.8	9.88	11.67	13
Sentence standard deviation in terms of no. unique content words	7.73	7.44	8.37	8	7.98	7.48	6	5	6.28	6.21	10	9	5.95	6.39	6.31	6.07
Entropy																
Word entropy	5.43	5.03	5.09	5.21	5.19	5.52	4.85	4.48	5.12	5.01	4.99	5.32	5.54	4.96	5.08	5.12
Character entropy	2.91	2.84	2.91	2.9	2.88	2.89	2.88	2.88	2.73	2.92	2.89	2.93	2.91	2.9	2.91	2.9
Word complexity																
Average distance between lemma and word stems (only content words)	1	1.38	1.12	1.17	1.05	1.17	0.86	1.09	1.09	1	1.27	1.2	1	1.06	1.15	1.05
Average distance between words and corresponding stems (only content words)	1.6	2.26	1.82	1.91	1.74	1.86	1.67	2.05	1.83	1.7	2.03	1.74	1.64	1.92	1.71	1.61
Semantics																
Average paragraph score	91	46	15	27	39	86	13	100	11	16	16	45	101	20	25	25
Paragraph score standard deviation	0	0	12	25	0	82	10	98	8	15	14	42	99	18	24	22
Average sentence score	2.24	3.15	2.08	2.37	1.17	3.37	1.67	3.81	0.94	1.26	2.18	2.57	3.22	2.06	2.33	2.16
Sentence score standard deviation	1.39	1.12	1.36	2.77	1.2	4.5	1.58	4.46	1.02	2.25	2.11	2.51	4.31	2.19	2.53	1.83

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Average relevance of top 10 keywords	1.5	1.42	1.19	1.49	1.15	1.66	1.13	1.63	1.17	1.08	1.44	1.39	1.7	1.2	1.36	1.37
Standard deviation of the relevance of top 10 keywords	0.24	0.52	0.18	0.32	0.32	0.28	0.25	0.29	0.22	0.58	0.51	0.32	0.23	0.37	0.26	0.21
Average sentence-paragraph cohesion (LSA)	0.42	0.6	0.62	0.72	0.35	0.55	0.58	0.54	0.49	0.18	0.74	0.58	0.54	0.62	0.72	0.58
Average sentence-paragraph cohesion (LDA)	0.46	0.63	0.67	0.74	0.43	0.66	0.68	0.51	0.54	0.74	0.76	0.62	0.65	0.66	0.75	0.66
Average intra-paragraph cohesion (LSA)	0.27	0.45	0.22	0.28	0.19	0.24	0.22	0.3	0.12	0.22	0.25	0.29	0.21	0.23	0.28	0.21
Average intra-paragraph cohesion (LDA)	0.54	0.55	0.54	0.57	0.46	0.62	0.5	0.51	0.5	0.53	0.5	0.54	0.53	0.56	0.54	0.57
Average sentence adjacency cohesion (LSA)	0.27	0.45	0.22	0.28	0.19	0.24	0.22	0.3	0.12	0.22	0.25	0.29	0.21	0.23	0.28	0.21
Average sentence adjacency cohesion (LDA)	0.54	0.55	0.54	0.57	0.46	0.62	0.5	0.51	0.5	0.53	0.5	0.54	0.53	0.56	0.54	0.57

Step 4. Conceptualize and visualize the key associations between the central concepts of all considered articles. Figure 3 depicts the central keywords from all articles, the size of each node is proportional to the relevance of each word and the links among concepts are determined as the average value of LSA and LDA semantic similarities.

Step 5. Evaluate semantic similarities (LSA & LDA cohesion-based) between articles in order to observe potential associations. Based on the used keywords, semantic similarities are computed between all pairs of articles, generating the relatedness graph presented in Figure 4, as well as the pairs of most similar articles.



Top Similar Articles

Article	Article	Semantic Similarity Score
Articol 3	Articol 4	0.89
Articol 3	Articol 6	0.84
Articol 4	Articol 6	0.84
Articol 15	Articol 6	0.79
Articol 13	Articol 15	0.79
Articol 15	Articol 16	0.79
Articol 16	Articol 6	0.79

Figure 3. Global concept map of keywords extracted from all articles

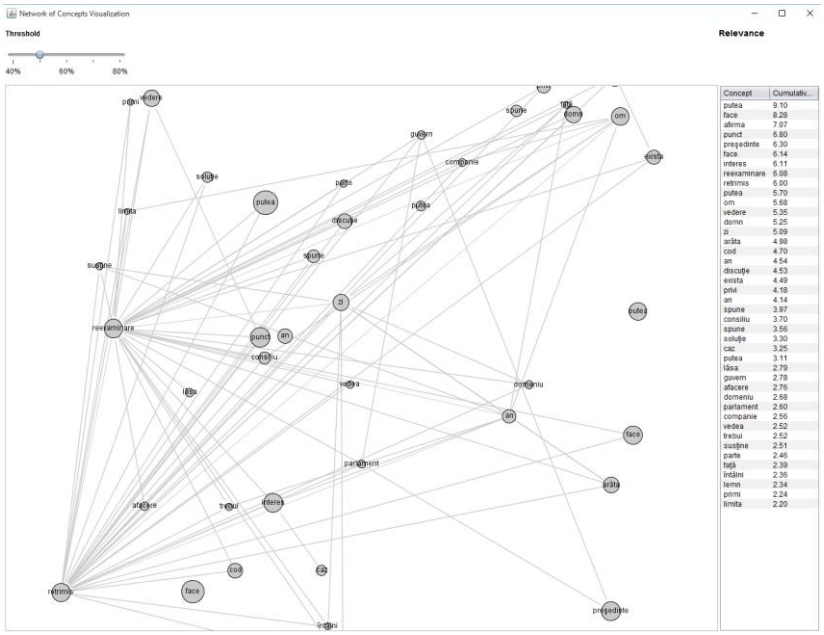


Figure 4. Semantic relatedness between all articles

Step 6. Perform a hierarchical clustering of articles based on the average linkage between normalized textual complexity indices, in order to identify similar writing styles. Table 2 together with Figure 5 present the stages of the agglomerative clustering algorithm in which the most similar articles in terms of complexity indexes are grouped together. Remarkable join stages are between articles 15 & 16 (stage 1) and 3 & 4 (stage 3) which are also very similar in terms of used concepts and vocabulary, not only writing style.

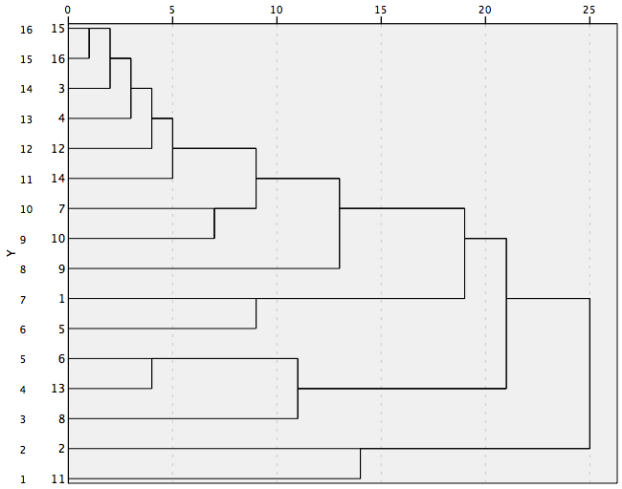


Figure 5. Dendogram using average linkage between articles' complexity indices

Table 2. Coefficients of similarity between articles generated by the agglomerative clustering algorithm

Stage	Similar articles combined		Coefficients
	Article ID 1	Article ID 2	
1	15	16	8.615
2	3	15	13.646
3	3	4	15.426
4	3	12	19.278
5	6	13	20.973
6	3	14	24.882
7	7	10	31.255
8	1	5	37.261
9	3	7	37.337
10	6	8	44.385
11	3	9	49.939
12	2	11	53.683
13	1	3	68.609
14	1	6	76.821
15	1	2	90.955

5. Conclusions and future work

Our pilot study performed on the daily *Adevărul*, covering several articles signed by different journalists, highlighted different features reflected in textual complexity indices adapted for Romanian language. For the moment, the textual complexity indices denote a rather high similarity in terms of writing style, a normal trait taking into consideration the semantic similarity between the articles, the same genre and the same target audience. Although the statistics are satisfactory, it is premature to advance some firm conclusions about the accuracy of the obtained data and more in-depth analyses covering a broader input dataset will be considered.

In addition, as an extension of the current study we envision the automatic classification of anonymous readers using their reactions (comments) in *Adevărul* forum, as the lecture of online articles by various users is in most cases superficial. For the time being, the selected corpus was too small and a classification of anonymous readers based on their answers and active participation was inappropriate. Therefore, we intend to extend our cohesion-centred evaluation by introducing also the comparison between articles and the corresponding forum comments. In the end, our classification would enable a more comprehensive profiling of anonymous readers and a better adaptation of journalistic writing to the targeted audience.

References

Blei, D. M., Ng, A.Y., and Jordan, M.I. (2003). Latent Dirichlet Allocation. In *Journal of Machine Learning Research*, 3, pp. 993-1022

Dascălu, M. (2014). Analyzing discourse and text complexity for learning and collaborating, Studies in Computational Intelligence, Vol. 534, Switzerland: Springer.

Dascălu, M., Dessus, P., Bianco, M., Trausan-Matu, S., and Nardy, A. (2014). Mining texts, learners productions and strategies with ReaderBench. In A. Peña-Ayala (Ed.), *Educational Data Mining: Applications and Trends*, Switzerland: Springer, pp. 335–377.

Dascălu, M., Dessus, P., Trausan-Matu, S., Bianco, M., and Nardy, A. (in press). ReaderBench, an Environment for Analyzing Text Complexity and Reading Strategies. In *16th Int. Conf. on Artificial Intelligence in Education* (AIED 2013). Memphis, USA: Springer.

Dascălu, M., Stavarache, L.L., Dessus, P., Trausan-Matu, S., McNamara, D. S., and Bianco, M. (2015). Predicting Comprehension from Students' Summaries. In *17th Int. Conf. on Artificial Intelligence in Education* (AIED 2015), Madrid, Spain: Springer, pp. 95–104.

Dascălu, M., Trausan-Matu, S., and Dessus, P. (2012). Towards an integrated approach for evaluating textual complexity for learning purposes. In E.

Popescu, R. Klamma, H. Leung & M. Specht (Eds.), *11th Int. Conf. in Advances in Web-Based Learning* (ICWL 2012), Sinaia, Romania: Springer, pp. 268–278.

Gîfu, D. and Cristea, D. (2013). Monitoring and predicting journalistic profiles in Computational collective intelligence: technologies and applications (including subseries Lecture Notes in *Artificial Intelligence and Lecture Notes in Bioinformatics)*, Volume 8083 LNAI, 2013, C. Bădică, N.T. Nguyen and M. Brezovan (eds.), Springer-Verlag Berlin Heidelberg, pp. 276-285.

Gîfu, D. and Cioca, M. (2013). Online Civic Identity. Extraction of Features in *Procedia – Social and Behavioral Sciences*, vol. 76/15, edited By Emanuel Soare, ELSEVIER, pp. 366-371.

Gîfu, D., Stoica, D. and Cristea, D. (2013). Virtual Civic Identity in *Proceedings of The 9th International Conference Linguistic Resources and Tools for Processing The Romanian Language*, ConsILR-2013, 16-17 May 2013, Miclăușeni, Elena Mitocariu, Mihai Alex Moruz, Dan Cristea, Dan Tufiș, Marius Clim (eds.), "Alexandru Ioan Cuza" University Publishing House, Iaşi, pp. 139-148

Grosz, B.J., Weinstein, S., & Joshi, A.K. (1995). Centering: a framework for modeling the local coherence of discourse. Computational Linguistics, 21(2), pp. 203–225.

Hayes, D. P., Wolfer, L. T., & Wolfe, M. F. (1996). Sourcebook simplification and its relation to the decline in SAT-Verbal scores. *American Educational Research Journal*, 33, pp. 489–508.

Landauer, T.K. and Dumais, S.T. (1997). A solution to Plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge. In *Psychological Review*, 104(2), pp. 211-240

Manning, C.D., & Schütze, H. (1999). Foundations of statistical Natural Language Processing. Cambridge, MA: MIT Press.

McNamara, D.S., Graesser, A.C., & Louwerse, M.M. (2012). Sources of text difficulty: Across the ages and genres. In J. P. Sabatini, E. Albro & T. O'Reilly (Eds.), *Measuring up: Advances in how we assess reading ability*, Lanham, MD: R&L Education, pp. 89–116.

Nelson, J., Perfetti, C., Liben, D., & Liben, M. (2012). Measures of text difficulty: Testing their predictive value for grade levels and student performance. Washington, DC: Council of Chief State School Officers.

Page, E. (1966). The imminence of grading essays by computer. Phi Delta Kappan, 47, pp. 238–243.

Page, E. (1968). Analyzing student essays by computer. *International Review of Education*, 14(2), pp. 210–225.

Powers, D.E., Burstein, J., Chodorow, M., Fowles, M.E., & Kukich, K. (2001). Stumping e-rater®: Challenging the validity of automated essay scoring. Princeton, NJ: Educational Testing Service.

Shannon, C.E. (1948). A Mathematical Theory of Communication. *The Bell System Technical Journal*, 27, pp. 379–423 & 623–656.

Shannon, C.E. (1951). Prediction and entropy of printed English. *The Bell System Technical Journal*, 30, pp. 50–64.

Slotnick, H. (1972). Toward a theory of computer essay grading. *Journal of Educational Measurement*, 9(4), pp. 253–263.

Stenner, A. J., Koons, H., & Swartz, C. W. (2010). Text complexity and developing expertise in reading. Durham, NC: MetaMetrics, Inc.

Trausan-Matu, S., Dascalu, M., & Dessus, P. (2012). Textual complexity and discourse structure in Computer-Supported Collaborative Learning. In S. A. Cerri, W. J. Clancey, G. Papadourakis & K. Panourgia (Eds.), *11th Int. Conf. on Intelligent Tutoring Systems* (ITS 2012), Chania, Grece: Springer, pp. 352-357.

Teh, Y.W., Jordan, M. I., Beal, M.J., & Blei, D.M. (2006). Hierarchical Dirichlet Processes. *Journal of the American Statistical Association*, 101, pp. 1566–1581.

Tufiş, D., Radu, I., Bozianu, L., Ceauşu, A., and Ştefănescu, D. (2008). Romanian Wordnet: Current State, New Applications and Prospects. In *Proceedings of the 4th Global WordNet Conference*, GWC-2008, Szeged, Hungary, pp. 441-452.

van Dijk, T.A., & Kintsch, W. (1983). Strategies of discourse comprehension. New York, NY: Academic Press.

Vygotsky, L.S. (1978). Mind in society. Cambridge, MA: Harvard University Press.

Wresch, W. (1993). The imminence of grading essays by computer - 25 years later. Computers and Composition, 10(2), pp. 45–58.

IMPLEMENTATION OF A SIMPLE WEB SEARCH ENGINE

DIANA-ALEXANDRA SAVELUC

Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iași

alexandra.saveluc@info.uaic.ro

Abstract

This paper proposes a software solution for implementing a web search engine, considering the following aspects: dynamic page content, large number of websites in the world and the need of a fast response (usually in less than half of a second). A web search engine is a program that has two important features: indexing web pages, storing and providing information from these lists of pages corresponding to a given query. In this context, for calculating the page importance, we present comparative studies regarding the use of Google Page Rank algorithm and the On-line Page Importance Computation algorithm. This application allows users to realize queries in a database with web pages, returning them as a response a list of pages ordered by their importance.

Keywords: web search engine, indexing, crawler, interrogation mechanism, page rank+.

1. Introduction

This paper describes the implementation of a search engine and presents an analysis of its architecture. Search engines have become very important tools due to the very high growth rate in the number of available websites, reaching from one website in 1991 to nearly 1 billion in 2014. These tools act as filters for very large volume of available data. A good search engine allows the user to easily and quickly obtain information relevant to them without having to navigate on irrelevant sites.

But the search engine result page is computed based on several factors including the proximity of the search terms to one another, the frequency of the query terms within the page and the pages importance. There are many algorithms that compute the importance of web pages based on the number of links between them and their quality. These include PageRank, the first algorithm used by Google and one of the best known in this area, and Online Page Importance Computation, an algorithm that works online and does not require storing references from pages In section 4 we present a comparative study between these two algorithms and we show that both generate about the same results, but one is more optimal considering the execution time. Although the results page is generated in a very complex way, often a search engine does not provide the expected results and can contain many sponsored links.

For this purpose we have designed a module that allows the user to customize the results by modifying the score pages through a user friendly interface. Thus search

engine result page will be personalized and the probability that it contains irrelevant links in the top of the list will decrease.

Implementing an entire search engine brings many challenges due to the characteristics of web space: large amounts of data, high rate of change and dynamically generated pages. We tried to find an optimal way to implement three application modules: crawler, indexer and query mechanism, so they can meet the requirements listed above.

The paper is organized as follows: Section 2 presents some representative works important for this survey, Section 3 describes the technology required to implement a web crawler and depicts our web application. Section 4 presents comparative studies between Google Page Rank algorithm and online Page Importance Computation. Finally, conclusions and discussions with reference to the obtained results will be mentioned in the last section.

2. Background

The first search engine exists since the beginning of the internet, in December 1990. The first tool for searching content (as opposed to users) on the Internet was Archie1 (Emtage and Deutsch, 1992). All these tools have been improved continuously and Google was a company that experienced a very large development, especially after 2000. They introduced the concept of PageRank, in which the anatomy of a Search Engine was explained (Brin and Page, 1998).

But lately, many researchers have been interested in this field (Abiteboul *et al.*, 2003; Najork, 2009; Lewandowcki, 2012, etc.) due to the large increase of available data on the internet, the number of users, and the fact that the implementation of this tool brings many challenges (Abiteboul *et al.*, 2011). They presented the structure of a search engine, the concept of inverted file or Fagin's threshold algorithm that allows top-*k* answering queries. We will use the theoretical foundations presented in their article for our implementation, shortly described below.

3. Technology

In order to create the data that will be used to provide search results, we implemented a web crawler, a software program that browse and downloads web pages that are sent to the indexer.

3.1 Web crawler

This module aims to identify a large number of pages available on the web and download them. Browsing the Web space is a difficult task and an important part of a search engine. A web crawler must effectively solve the following problems: (1)

[1] *Archie* is the first Internet search engine used for indexing FTP archives, allowing people to find specific files - http://web.archive.org/web/20070621141150/http://isrl.uiuc.edu/~chip/projects/timeline/1990archie.htm

identify duplicate pages; (2) establish the next link that will be visited, and (3) decide how often have the sites that were already indexed to be parsed again.

To visit the web addresses available we follow the steps: start at a given URL (or set of addresses), index the Web document and parse to reveal hyperlinks. We indexed the addresses and repeated the process for each hyperlink found. Since it was impossible to index the entire web address space, there is a limit of the number of pages that can be processed.

This problem is similar to a graph traversal problem and can be solved either through a Breadth-First Search (all pages marked by another page are indexed before the links they contain are analysed), either through a Depth-first Search (page which contains a hyperlink is indexed and its links to other pages are indexed and analysed to discover new addresses). Since the possibility to enter thus an infinite loop exists, a mix of the two methods could be a solution: a Depth-First Search only for links that are good candidates. Next we treat another problem encountered in creating a web crawler: the detection of duplicate pages. There are two types of such pages: identical and near-duplicates.

Identical pages are easily identified by using a hash function. To identify near-duplicates pages we compute the edit distance between two documents. The edit distance is the number of basic changes (adding or deleting words or characters) that you need to perform to get a document from another. This may be computed using a dynamic programming algorithm with the complexity O($m*n$), where m, n are the sizes of the two documents. But this is not a viable solution for a collection of documents because we will have to repeat this process for any two documents. Therefore we use a sequence of fixed-length tokens. For each document d and d' we compute two sets of tokens S and S'. These tokens will be used to compute Jaccard's coefficient as follows:

$$J(S,S') = \frac{|S \cap S'|}{|S \cup S'|}$$

and will be the degree of similarity between two documents (1 if the two documents are identical).

3.2 Search index and inverted file

We implemented a search index called „Inverted Files". Search index resembles the index at the end of a book and contains all the words that appear in web pages, listed alphabetically, and each word is related to a list of references in those web pages.

Figure 1 shows the structure of an inverted file. The file size depends on the number of documents used. Here, t_1 is a term with a huge number of appearances in documents, t_m is the term with a small number of appearances, and n_i is the number of documents containing the term t_i. Each entry e in the list contains the index of the document containing the term (d_i represented by an unsigned integer) and term position in the document (w_i).

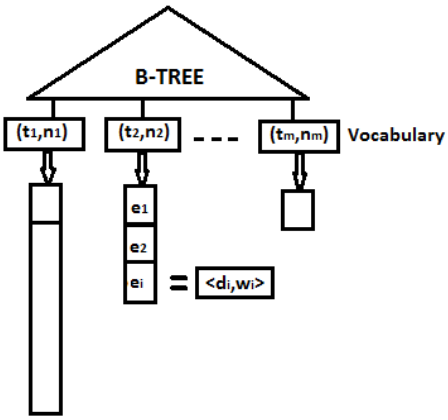


Figure 1: Structure of an inverted file

To obtain a set of documents (IDs) containing a term `t`, we could easily iterate through the list associated with the term `t`. But clearly, some documents are more relevant than others: for this reason we measure the relevancy of a page's content by assigning a certain weight to the occurrence of a term in a document.

One method of finding this weight is TF-IDF - "term frequency - inverse document frequency". A term frequency (`tf`) represents the number of occurrences of the term `t` in document `d`, divided by the total number of terms in `d`:

$$\mathrm{tf}(t, d) = \frac{n_{t,d}}{\sum_{t'} n_{t',d}} \quad (1)$$

Inverse Document Frequency (IDF) measures the importance of a term for a collection of documents D. It is obtained by dividing the total number of documents by the number of documents in which `t` appears:

$$\mathrm{idf}(t) = \log \frac{|D|}{\left|\{d' \in D \,|\, n_{t,d'} > 0\}\right|} \quad (2)$$

The mathematical formula for calculating the weight of a term in a document `d` is:

$$\mathrm{tfidf}(t, d) = \frac{n_{t,d}}{\sum_{t'} n_{t',d}} \cdot \log \frac{|D|}{\left|\{d' \in D \,|\, n_{t,d'} > 0\}\right|} \quad (3)$$

This weight and index entries will be added to the document structure.

Example: Consider a set of documents d_1 ... d5. Entry d_1 / 11 / 0.7 means that the term appears in the document d_1 11 times and has a weight of 0.7.

3.3 Query processing mechanism

This module processes a user query and applies a search algorithm to provide a list of web pages as a result. Query processing is performed in two steps: in the first step we take from the search index some information about relevant web pages that match the keywords in the user query and in the second phase we will create a ranking of results starting from the most relevant down.

Consider a query like "t_1 and t_2... and t_n" and K a fixed number of documents that we want to show. We presented two algorithms based on an inverted file and top-K queries.

The first algorithm (ranked queries base algorithm). Suppose that lists $Lt_1, ..., Lt_n$ are sorted by the document ID. We will conduct a parallel search in all lists $Lt_1, ... , Lt_n$ to find a list of the form $[d_i(1), ... d_i(n)]$ (a document d_i containing all terms). We note $s(t, d)$ - the weight of the term t in a document d (using TF - IDF) and $g(s_1...s_n)$ – a monotonic function that computes the overall score for the document based on the weight of each term. The overall score of the document $Wi = g(s(t_1, d_i)...s(t_n, d_i))$ is added to a vector of pairs $[di\ Wi]$. Once we complete the search, we sort the vector according to the overall score and show the top K documents from that vector.

The effectiveness of this algorithm depends on the user's query semantics. If required the appearance of each term in the outcome document, the search can be stopped when one of the lists have been completed. But for a query that contains the OR operator, resulting documents can score higher overall even if some terms are missing. To solve this problem, we use the Fagin's threshold algorithm. Each index contains, besides lists of identifiers in ascending order, lists of weights sorted in ascending order.

Fagin's threshold algorithm

1. Let R, the result set, be the empty list
2. For each 1<= i<=n
   a. Retrieve the document d(i) containing term ti that has the next largest s(ti,d(i)).
   b. Compute its general score gd(i) by retrieving all s(tj, d(i)) with j != i.
   $$g_{d(i)} = g(s(t_1, d^{(i)}), \dots, s(t_n, d^{(i)}))$$
   If the query is a conjunctive query, the score is set to 0 if some s(tj ,d(i)) is 0.
   c. If R contains less than K documents, add d(i) to R. Otherwise, if gd(i) is larger than the minimum of the scores of documents in R, replace the document with minimum score in R with d(i).
3. Let
$$\tau = g(s(t_1, d^{(1)}), s(t_2, d^{(2)}), \dots, s(t_n, d^{(n)}))$$

4. If R contains at least K documents, and the minimum of the score of the documents in R is grater than or equal to π, return R.
5. Redo step 2.

4. Page rank

Although "term frequency inverse term frequency" adds weight to a document based on keywords, it does not distinguish between important pages (reliable) and those that may contain irrelevant information. Therefore, we decided to organize the pages into a graph, to compute a score for each document.

PageRank can be defined as the probability `pr(i)` that a user reaches a specific page from another separate page. This is computed for all pages in the graph and is independent of any query. PageRank will be used to update the weights of our inverted file as follows: `weight(t,d) = tfidf(t,d) × pr(d)`.

PageRank is an important algorithm, being used by many search engines and crawlers to display lists of results. A crawler can use PageRank as one of a number of importance metrics it uses, to determine which link to visit during a crawl of the web and a search engine will use it to rank websites in search engine results. Page rank computation in a graph with a very large dynamic content is a topic that has captured much attention in the context of search engines.

5. Mathematical model

We can represent web space as a directed graph `G`, where web pages are nodes and links from one page to another are arrows. We say that the graph is called weakly connected if, replacing all of its directed edges with undirected edges, a connected (undirected) graph is produced. A directed graph is “strongly connected” if, it contains a directed path from `u` to `v` and a directed path from `v` to `u` for every pair of vertices `u`, `v`. We will consider the graph structured as a matrix.

In general, let `G` be a directed graph with n nodes. `G` can be represented as a matrix `L[1..n,1..n]` as follows

• any `i, j` and `[i, j]> = 0, 1 <= i; j <= n`

• `L [i, j]> 0` if there is an arrow between `i` and `j`.

Computation of the page importance is done inductively. If the graph contains `n` nodes, the importance of the page is represented as a vector `x` in `n` dimensions. We present three examples in which the importance is computed inductively using the equation: `k+1 =L*xk`. A page is important if it succeeds an important page. In this case we set:

`L [i, j] = 1` if there is an edge between `i` and `j`.

For the case of a random traversal we set `L [i, j] = 1 / d [i]` if there is an edge between `i` and `j`.

We will decide that a page is important if it succeeds or precedes an important page. We set `L [i, j] = 1` if there is an edge between `i` and `j` or from `j` to `i`.

One of the algorithms used to compute the importance of pages is OPIC (*Online Page Importance Computation*) which operates online and consumes fewer resources than its predecessor's algorithms. This algorithm does not require storing an array of references of pages and is called online because recalculates a page's importance as long as the web is analysed.

5.1 Algorithm description

For each page will keep two values. The first is called "cash" and it is initialized to `1 / n` (in a graph with `n` edges). While the algorithm is running, the cash will retain newly discovered information about the page ("the cash" obtained by processing the last page). We will also retain the value of "history" of a page and a global variable `G`.

5.2 On-line Page Importance Computation

```
for each i let C[i] := 1/n ;
for each i let H[i] := 0 ;
let G:=0 ;
do forever
begin
choose some node i ;
%% each node is selected
      H[i] += C[i];
      %% single disk access per page
      for each child j of i,
            do C[j] += C[i]/out[i];
            %% Distribution of cash
            %% depends on L
            G += C[i];
            C[i] := 0 ;
End
```

At every step we estimate the PageRank of a page as `(H + C) / (G + 1)`.

Web application

In order to implement a search engine we have developed a web application that gives the user the ability to perform queries based on keywords. The application

consists of 4 modules: the web crawler; the query mechanism; dictionary words and a web application that facilitates user interaction.

The crawler will receive as parameters the address from which we start browsing the web, and the number of pages that we want to go through. Because the Web is structured as a graph, the crawler will perform Breadth-First Search from the URL set by the administrator and will add the addresses discovered in the database. Processing a newly discovered page will involve the extraction of page tags like `"<a href \\ \\ s + s * = \\ s * \"? (. *?) [\ "|>]"` by skipping JavaScript code, and over tags that contain `"#"` (references to the current page). Links found will be added to the table `"Link"` from the database, if are not already added, and the process is repeated until the number of pages found is equal to that received as an input parameter. This module will compute also the page importance and add this value to the database.

The module *Dictionary Words* seeks to create a dictionary including all the words in the pages inserted into the database. The words will be retained in a binary tree because operations will be executed in the worst case in `O(log n)`. For each page in the database we will extract its contents and create a list of words from which we will remove prepositions and determinants. For each word in the list we compute the weight and we will add this value to the corresponding position in the binary tree as inputs (document, weight). Thus, every word that is retained in the tree will have an associated list of entries. After creating the dictionary, we will serialize it to be stored more easily.

Query mechanisms receive a user query (several words) and apply a search algorithm to provide a list of web pages as a result. In the first step we took from the dictionary lists of documents associated with the query words. We performed a parallel search in these lists to find those documents containing all the words in our query. In a second step we produced a ranking of the results from the most relevant down.

Web application interface is divided into two sub-modules: one for the user and one for the administrator. Regarding user dedicated sub-module: the user will enter a query (word list) and will receive in response a list of URLs. The task of returning the result to the user is accomplished by the third module of the application (query mechanism). If the administrator successfully logs in, then he has to configure a set of features to set the search engine. The administrator can fill in the database with URLs using the Web Crawler option and can edit the pages obtained. Thus, if he would like to have some pages hidden to the user, he will remove them from the list of addresses. Example: if the site administrator will want the search engine to be dedicated to children, the administrator will begin crawling the web from a directory of websites for children, and delete those pages that are not appropriate to their age. After filling the database, the administrator will be able to compute and edit the Page Rank for each page. Thus, he can decrease the page rank of those pages that do not respect copyrights, or those containing irrelevant data or too many advertisements.

Regarding the technologies, we use Java on the backend because it is portable and runs on almost any hardware and software platform. The data were stored using MySQL and the application interface was developed using JSP, HTML5, CSS3, JavaScript and JQuery to facilitate user interaction with the application.

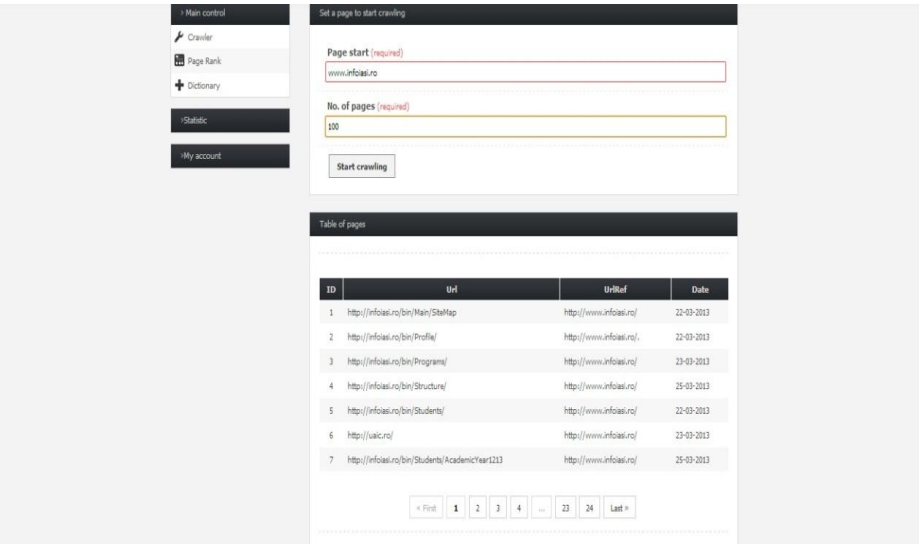


Figure 2: Crawler functionality of admin's page

6. Case study

Below we present a case study: the time necessary for crawling the web and ranking pages by their importance. Initially, for the web crawler implementation we used a Breadth-First Search algorithm and for finding the importance of pages – "Google PageRank". The complexity of such an algorithm is `O(n+m)`, where `n` is the number of nodes in the graph and `m` – the number of edges, so we can say that BFS algorithm is optimal.

The Google PageRank algorithm operates offline: after all pages have been indexed by the crawler, a G matrix (Google Matrix) is built as follows:

$$G=d*H+((1-d)/N)*U$$

where `H` is the matrix of links between pages and is defined by:

`H[i,j]= {0`, if there is no edge between `i` and `j`; 1/out degree of node `j`, otherwise}

`N` = number of nodes in the graph;

`U` = unit matrix of size `N*N`;

D = the damping factor probability of a user to follow links on the page without realizing jumps to other pages (in this case we consider d=1).

The vector that will retain PageRank sites will be the x column vector satisfying the property: x = G*x. The time complexity of this algorithm is $O(N^2)$, so for the entire process it will be $O(n+m)+O(N^2)=O(N^2)$.

Table 1: Page rank computed based on Google Page Rank algorithm

ID	URL	PageRank
1	http://dsc.discovery.com	0.17
2	http://dsc.discovery.com/tv-shows	0.09
3	http://dsc.discovery.com/videos	0.09
4	http://games.dsc.discovery.com/	0.09
5	http://dsc.discovery.com/adventure	0.09
6	http://dsc.discovery.com/cars-bikes	0.09
7	http://dsc.discovery.com/gear-gadgets	0.09
8	http://dsc.discovery.com/tv-shows/shark-week	0.09
9	http://news.dsc.discovery.com	0.09
10	http://dsc.discovery.com/a-curios-discovery	0.09

Table 1 shows the results of the crawler having as starting point the address: dsc.discovery.com and 10 pages being indexed. Once we have indexed the pages, their importance is computed offline. Indexing pages by crawler was done in 3,357 milliseconds and for computing their importance another 83,304 milliseconds were required (a total of 86,661 milliseconds).

We noted that the execution time required is quite high seen the number of pages indexed. Therefore, we tried to improve the application so the time required for this process to be lower. In this regard, we used the OPIC algorithm[2] to compute the page importance. It works online and consumes fewer resources than Google PageRank (no need to store an array of references of pages). The advantage of this algorithm is that it will calculate the page rank while the web space will be covered by the crawler. Thus the complexity of the two processes will be $O(n+m)$.

The following table shows the results of the new crawler with the same input. The crawler has indexed these addresses and computed their importance in 12,511 milliseconds. The ratio of page importance remained about the same, but the execution time decreased with approximately 80%.

[2] On-Line Page Importance Computation
http://www2003.org/cdrom/papers/refereed/p007/p7-abiteboul.html

Table 2: Page rank computed with the OPIC algorithm

ID	URL	Cash	History	PageRank
1	http://dsc.discovery.com	0.060	0.100	0.078
2	http://dsc.discovery.com/tv-shows	0.007	0.100	0.053
3	http://dsc.discovery.com/videos	0.006	0.102	0.053
4	http://games.dsc.discovery.com/	0.006	0.103	0.053
5	http://dsc.discovery.com/adventure	0.004	0.103	0.053
6	http://dsc.discovery.com/cars-bikes	0.003	0.104	0.053
7	http://dsc.discovery.com/gear-gadgets	0.002	0.105	0.053
8	http://dsc.discovery.com/tv-shows/shark-week	0.001	0.106	0.053
9	http://news.dsc.discovery.com	0.001	0.107	0.053
10	http://dsc.discovery.com/a-curios-discovery	0.000	0.100	0.049

7. Conclusions and future work

In this paper we described the implementation of a simple search engine. We presented a comparative study between two algorithms that compute pages importance and showed why one is prefered over the other.

The main advantage of this application is that the user is able to edit the content of the database where the search will be performed (the user can delete/edit URLs or their PageRank).

We think that there are some very interesting directions we would like to investigate further. The first one is to consolidate the actual database, to do some tests and analyze how quickly our solution will give an answer in cases of large amounts of data indexed, and how we can improve the response time. The second improvement is related to the implementation of the application available for Android and iOS operating systems.

References

Abiteboul, S., Manolescu, I., Rigaux, P., Rousset, M.-C., Senellart, P. (2011). Web data management, Cambridge: University Press.

Abiteboul, S., Preda, M., Cobena, G. (2003). Adaptive on-line page importance computation. In *Proceedings of the 12th international conference on World Wide Web*. Budapest, Hungary: ACM. pp. 280–290.

Brin, S. and Page, L. (1998). The Anatomy of a Large-Scale Hypertextual Web Search Engine. In: *Seventh International World-Wide Web Conference* (WWW 1998), April 14-18, Brisbane, Australia.

Emtage, A., and Deutsch, P. (1992). Archie - An Electronic Directory Service for the Internet. In *Proceedings of the Winter USENIX Conference*, Usenix Association, Berkely, San Francisco, CA, pp. 93-110.

Lewandowski, D. (2012). New perspectives on Web search engine research. In *Web Search Engine Research*, Emerald Group Publishing (http://books.emeraldinsight.com/display.asp?K=9781780526362).

Najork, M. (2009). Web Crawler Architecture Entry. In *Encyclopedia of Database System.*

CHAPTER 6
MORPHOLOGY AND SYNTAX

REGENERATION OF CULTURAL HERITAGE: PROBLEMS RELATED TO MOLDAVIAN CYRILLIC ALPHABET

CONSTANTIN CIUBOTARU, SVETLANA COJOCARU,
ALEXANDRU COLESNICOV, VALENTINA DEMIDOV,
LUDMILA MALAHOV

Institute of Mathematics and Computer Science, Moldavian Academy of Sciences

{svetlana.cojocaru, constantin.ciubotaru, kae, valentina.demidov, mal}@math.md

Abstract

The paper is concerned with techniques for creation of linguistic resources for historical Romanian. The problems related to the Moldavian Cyrillic alphabet for 1951–1991 in Bessarabia (Moldavian SSR) are discussed. The Romanian words transliteration from Cyrillic to the Latin alphabet is studied. The transliteration rules for converting (descyrillization) of words in Romanian language written in Cyrillic to their equivalent written in Latin script are elaborated and motivated. A special attention is paid to cases that present ambiguities. Descyrillization of the literature from this period would also permit to inject the most valuable parts of this heritage in the Romanian cultural life.

Keywords: recognition of historical printed text, transliteration, descyrillization, electronic lexicography, language development.

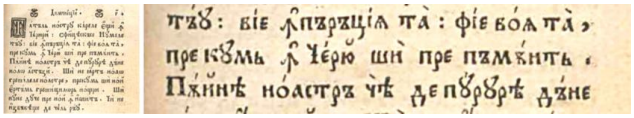
1. Restoration of Romanian Cyrillic texts

Historically, three types of Romanian Cyrillic scripts were used (Boian *et al.*, 2014). Romanian Cyrillic in 47 letters existed in Romania (including Bessarabia) till the middle of 19 century. Bessarabia used its variant based on Russian civil script till the 1 quarter of the 20 century. Then Moldavian Cyrillic script was used in Transnistria and, later, Bessarabia (Moldavian ASSR, Moldavian SSR) till 1989. The latest variant is still used in Transnistria. Samples of these scripts are shown in Fig. 1.

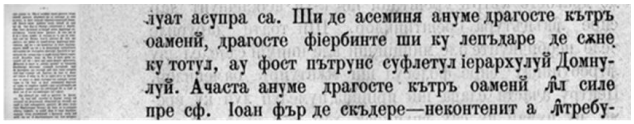
We take years 1951–1989 and restrict ourselves by the scientific and technical texts. We are interested in preparation of manuals, textbooks, and monographs for their re-edition. Electronic linguistic resources created for the mentioned task can be used at research on development of the Romanian language.

Regeneration of Romanian Latin text of the book consists of the following stages: scan and image pre-processing of Romanian Cyrillic texts; recognition (OCR); manual correction of recognition errors in the Cyrillic text; transliteration of Romanian text from Cyrillic to Latin script; editing of transliterated text; preparation of camera ready manuscript.

1



2



3

Минунат ши де некупринс есте очеанул математик. Дар пуцинь сынт чей каре избутеск сэ ажунгэ ушор ын стрэфундурile луй ши сэ-й куноаскэ ынтряга фрумусеце аскунсэ аколо. Дакэ, ынсэ, ай авут норокул сэ те куфунзь ын лумя каптивантэ а креацией математиче, е апроапе ку непутинцэ сэ-ць стэвилешть доринца де а арэта алткуйва, мэкар ынтр'ун фел, сплендида структурэ а математичий. Яр де ну ай посибилитатя сэ атражь дупэ …

Figure 1. Samples of Cyrillic scripts for Romanian

Scan and image pre-processing. We experimented with book (Acvariu, 1988). The print quality was quite satisfactory. There were no special difficulties during scan. Before the OCR stage, the images were pre-processed by the ScanTailor program from http://scantailor.org/.

OCR. We used the FineReader OCR engine to recognize Romanian Cyrillic text of the respective book. FineReader recognizes all Unicode characters in many typefaces. We defined the corresponding subset of Unicode as a new "user language". We took the Russian alphabet as the base, deleted letters ***ё***, ***щ***, ***ъ***, and added letter ***ӂ***. The last letter was introduced in 1967 for sound [dZ]. FineReader may be provided with a lexicon (word list). Lexicon is used at recognition of poorly printed words, and at the hyphenation processing.

Manual correction of recognition errors. The print quality was good so the hyphens elimination was mainly necessary on this stage. The following statistics of the text in Moldavian Cyrillic script was obtained for (Acvariu, 1988): 244 pages; approx. 364,000 characters (incl. blank spaces); approx. 61,300 words; approx. 8,050 different words.

Would the print (scan) quality worse, the word list in Cyrillic script could help.

We can produce the word list from the OCRed text, edit the list manually, and repeat OCR.

Obtaining Moldavian Cyrillic lexicon by transliteration of the modern Romanian word list from the Latin script is also possible, but equally difficult. For example, there are approx. 20 rules for letter ***і*** that can be mapped to ***и***, ***й***, ***ь***, ***ю***, ***я***, ***ы*** (Demidova, 2014). The corresponding software is now under development.

Transliteration of text is letter by letter conversion of words from one script to another, in our case, from Cyrillic to Latin.

Transliteration between Cyrillic and Latin scripts for Romanian is a complicated process due to irregularities of the Moldavian Cyrillic writing.

For example, possible mappings for letter ***я*** are ***я→еа***, ***я→іа***, and ***я→а*** (***функция→funcția***). A special program created for this process is described below.

Transliteration is impossible for several categories of words.

- **Proper nouns** are written in the Cyrillic script more or less against their pronunciation. In the Latin script, proper nouns from the languages that use the Latin alphabet keep their original writing in most cases (***Лоран→Laurent***, not ***Loran***). An interesting example is the toponymal ***Оялок→Oyapock*** (French) or ***Oiapoque*** (Portugal); it's a river that makes border between Brasilia and French Guiana. The opposite case present pure Cyrillic proper nouns that may be written with the Latin letters against their pronunciation (***Иван→Ivan***), or as the analogous Romanian nouns (***Иван→Ion***). It is also necessary to keep in mind traditional transcriptions, for example: Ukrainian ***Київ*** = Russian ***Киев*** = English ***Kiev*** or ***Kyiv*** = Romanian Cyrillic ***Киев*** = Romanian Latin ***Kiev*** = Romanian phonetic transcription from Ukrainian ***Kîiiv***. We see that the traditional Romanian transcription ***Kiev*** follows the Russian pronunciation, and uses ***K*** instead of ***Ch***.
- **Words of foreign origin** may keep their original orthography in Romanian: ***дизайн→design*** (not ***dizain***).
- **Some words** can have different orthography in Cyrillic: ***кыне→câine*** (*dog*; not ***câne***); ***ешире→ieșire*** (*exit*; not ***eșire***); ***сынт→sunt*** (a form of verb *to be*).

All these cases are processed by our program using the extendable dictionary of exceptions.

Editing of transliterated text. It was noted that the text does not fully correspond to the modern standard norms of the Romanian language. Therefore the additional lexical and stylistic editing (actualization) is necessary after transliteration.

Preparation of camera ready manuscript. This includes insertion of formulas and diagrams, and adaptation and insertion of artistic graphics. Manuals can contain a lot of equations and drawings. They should be retyped manually, because the problem of equations recognition is not solved till now. We should extract graphical items and convert them to formats suitable for further processing. Some pictures can include inscriptions that should be transliterated by manual correction of the image.

2. Problems related to Moldavian Cyrillic alphabet

The paper (Petic and Gîfu, 2014) describes an experiment for the creation of a parallel Romanian corpus automatically aligned when both Cyrillic and Latin variants of a text are available. Some additional peculiarities of the descyrillization

process were unveiled in this research. The pure transliteration “letter to letter”, “letter to letter combination”, and “letter combination to letter” for the text used in (Petic and Gîfu, 2014) covers 98.2% of words. The remaining 1.8% of words were edited as “word to word”, “word to word combination” and “word combination to word”. The following differences in the orthographic rules were noted: use of hyphen instead of apostrophe (***ынтр’о → într-o***); elimination of hyphen (***вре-ун → vreun***).

3. Detailed rules of transliteration (descyrillization)

The algorithm is based on a set of descyrillization rules that perform the process in automated mode. The proposed rules are oriented to the Cyrillic alphabet and the corresponding orthography as for 1951–1989.

We identify two types of rules: basic and complex.

The latest Cyrillic alphabet for Romanian contains 31 letters. 26 letters can be processed under the basic rules of transliteration. Five letters remain for complex processing, namely: ***г***, ***к***, ***ч***, ***ы***, ***я***.

The basic rules map each Cyrillic letter to one or more Latin letters in the context independent mode. This mapping is shown in Tab. 1:

Table 1. Basic Cyrillic-to-Latin mapping

Cyrillic	а	б	в	д	е	ж	ӂ	з	и	й	л	м	н	о	п	р	с	т	у	ф	х	ц	ш	ь	э	ю	’
Latin	a	b	v	d	e	j	g	z	i	i	l	m	n	o	p	r	s	t	u	f	h	ţ	ş	i	ă	iu	-

The remaining conversions are context dependent, or even include disambiguation. Descyrillization in these cases is performing under the following rules.

Rules for letter *г*:

- ***г→gh*** before ***е, и, я, ю, ь***;
- ***г→g*** otherwise.

Examples: ***гиочел → ghiocel*** (*galanthus* or *snowdrop*), ***грек → grec*** (*Greek*).

Rules for letter к:

- ***к→ch*** before ***е, и, ю, я***;
- ***кс, кз → x***, with exception of ***есземă*** (*eczema*) and derivatives;
- ***к→c*** otherwise.

Examples: ***кимие→chimie*** (*chemistry*), ***комун→comun*** (*common*), ***ксерокс→xerox***.

Rules for letter ч:

- ***ч→c*** before ***е, и, ь, я***;
- ***ч→ce*** before ***а***;
- ***ч→ci*** before ***о, у***;
- ***ч→ci*** at the word end;

- ***ч→ci*** before a consonant.

Examples: ***чай→ceai*** (*tea*), ***черк→cerc*** (*circle*), ***чубоцэ→ciubotă*** (*shoe*), ***чинч→cinci*** (*five*).

Rules with disambiguation. A more complicated case is transliteration of letter ***ы*** that is mapped to ***î*** or to ***â*** depending on many factors. We formulate the rules against the recommendations of the Romanian Academy.

1. ***ы→î*** in the following positions:
   1. at the beginning of a word: ***în*** (*in*), ***înainte*** (*forward*), ***încă*** (*more*), etc.;
   2. at the end of the word: ***coborî*** (*to descend*), ***hotărî*** (*to solve*), etc.;
   3. in the words that are formed with prefixes ***re-***, ***ne-***, ***pre-***, ***supra-***, etc. (≈1140) and a word that starts with letter ***î***: ***reînnoi*** (*to renew*), etc.
2. ***ы→â*** otherwise, that is, in the following positions:
   4. inside words: ***când*** (*when*), ***vânt*** (*wind*) etc.;
   5. a special case is the proper noun ***România*** and its derivatives: ***român***, ***românesc***, etc.

Transliteration of the letter *я* is extremely complicated. There are three possible mappings of this letter:

- ***я→ia***
- ***я→ea***
- ***ия→ia***

In spite of the fact that for this letter we have only three variants, the fixation of the appropriate variant is, unexpectedly, an extremely complicated problem from the algorithmic point of view. It seems almost impossible to present formal rules of the selection.

To unveil these complications, we performed statistical analysis of the Romanian Dictionaries DEX'09 and DOOM 2 using their electronic presentation in DEX-online (Dexonline), in the Latin writing. Hereinafter we will use abbreviation DEX to denote samples from both sources. To refine our estimations, we used the list of Romanian words by frequency (Frequency).

False identification of ***я*** is possible at the use of Latin writing for statistics. There are Romanian words where combinations ***ia*** and ***ea*** do not correspond to ***я*** in the Cyrillic writing, like ***диалог↔dialog*** (*dialogue*). This was taken into account during our analysis. The origin of words was also checked. We also converted all text to lowercase, removed duplicates and unnecessary information.

Word-lemmas in DEX shows almost equal distribution of ***я***: we have both ***ea*** (5460 cases) and ***ia*** (6004 cases). For each case, we analysed position of **ea/ia** (in the beginning of the word, at the end of the word, or inside the word), and collect information on the preceding and the following letters.

The statistics collected from DEX permitted to formulate several empiric rules that follow.

At the beginning of the word, combination ***ia*** (281 cases) prevails over ***ea*** (18 cases). Moreover, analysis of these 18 cases shows that they are of foreign origin (like ***earl***) and should be processed through the exception dictionary.

With **я** at the end of the word, we can formulate a rule with context dependence:

1. ***ия→ia***: 1760 cases, e.g., ***академия→academia*** (*academy*).
2. ***ея→eia***: 57 cases, e.g., ***кея→cheia*** (*key*)**.**
3. ***я→ea*** at the end of the word and after a consonant: 1625 cases, e.g., ***лумя→lumea*** (*the universe*, form with definite article at the end). There is a frequent exclusion ***abia*** (*just*).

Then we analysed word-lemmas from DEX with ***ea*** and ***ia*** inside words trying to detect context dependence. We checked combinations with all letters of the Romanian alphabet, both preceding and following the letter ***я***. As the result, we found that at least several letter combinations with ***я*** that permit to formulate clear rules.

In the case of **precedence**, we found that combinations of letters ***а***, ***е***, ***ф***, ***у***. ***ж***, ***к*** with the following ***я*** can be resolved more or less definitely.

The combination ***aea*** can be found in 33 words from DEX, ***aia*** in 509 words. Therefore, we should map ***ая→aia.*** Moreover, all combinations ***aea*** are found in words like ***althaea*** that is the Latin name of a plant (in fact, ***althœa***; it means that in this word ***ea*** will not be converted to ***я*** in the Cyrillic writing). Such words are exclusions for our program.

Exactly the same situation arises with the combination ***fea*** that can be found in 16 word-lemmas from DEX, while ***fia*** in 197 words. The basic rule will be ***фя→fia***. Sixteen exceptions are false; e.g., ***nimfeacee*** is a Latin name of a family of plants, and the dictionary shows the reading ***e-a***, not diphthong ***ea***.

The combination ***uia*** can be found in 317 words from DEX, while ***uea*** in 2 words. In this context the main rule is ***уя→uia*** with exceptions.

We will not discuss further rules with the precedence; the previous examples show techniques and problems of the analysis.

We analysed also the cases with ***ea*** and ***ia*** **followed** by a letter. Rules can be formulated for ***х, и, к, п, у*** after the letter ***я***.

The combination ***яп*** presents a very interesting case. ***iap*** can be found in 39 words from DEX, while ***eap*** in 129 words. The rule is ***яп→eap*** because all 39 supposed exceptions with ***iap*** are false, like ***diapazon*** or ***priapism***. They are marked in the dictionary with i-a (no diphthong here!). The same situation exists with the rule ***яу→eau***. The combination ***iau*** meets in words like ***semiautomatic*** where ***ia*** is not a diphthong because ***semi-*** is a prefix to word ***automatic***.

We will not show other rules of this kind to shorten the discussion.

We can also formulate a rule ***я→ea*** for transliteration of ***я*** in several suffixes after consonants: ***-ea*** (f. sg. from ***-el***); ***-eală***; ***-eață***; ***-ească*** (f. sg. from ***-esc***). Example: русяскэ→***ruseascĕ*** (adjective *Russian*, feminine form).

We can also use rule ***я→ea*** for the suffix that marks residency ***-ян*** (***-ean***) after most consonants. Example: ***молдовян→moldovean*** (*inhabitant of Moldova*).

The transliteration algorithm based on the sequential application of these rules was developed. The algorithm makes an attempt to produce the most probable variants of transliteration taking contexts into account. The transliteration program is written in Java and uses the Romanian linguistic resources (ELRR). The interface has two parallel windows for the source text in the Cyrillic writing and for the transliterated text. The transliterated text can be edited. A dictionary of exceptions is also used that can be dynamically replenished. The result can be saved as a Unicode text.

4. Testing the transliteration program

The program window with the text from (Acvariu, 1988) is shown on Fig. 2.

The tested program used: statistical approach; analysis of prefixes (≈1140: auto-, re-, supra-, etc.); exception dictionary for words like ***ешире→ieșire*** (*exit*), ***пыне→pâine*** (*bread*), proper nouns, foreign words, etc. We have not integrated a spellchecker for auto-correction into the current version of the program but we plan this.

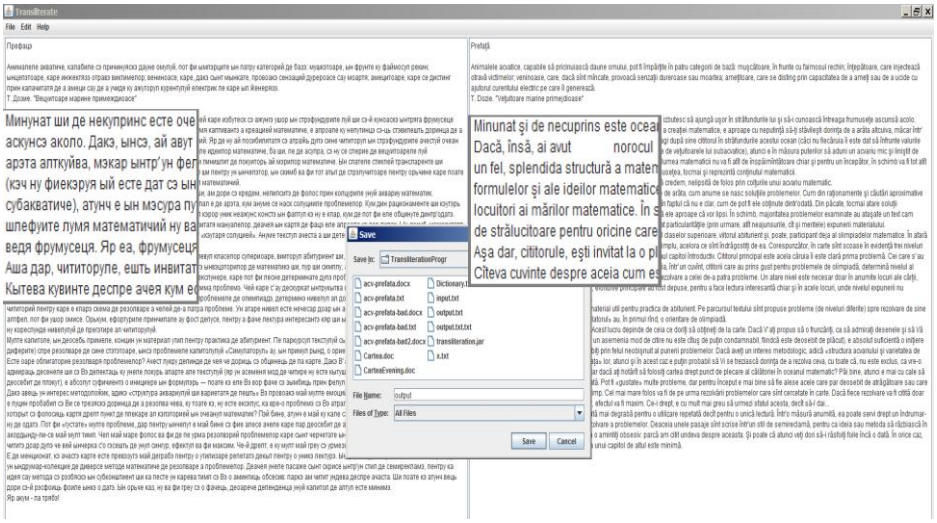


Figure 2. The transliteration program at work

For example, we get 6 errors ≈ 0.15% for juridical text of 6 pages and 4071 words.

This makes one error on a page. The same result was obtained with scientific and fiction text. All errors were of the type ***я→ia*** (instead of ***ea***), e.g., ***примяскэ→primiască*** (should be ***primeascĕ***).

5. Conclusions

The formulated rules became a base for development of descyrillization algorithm for Romanian. We are to note that the specific of language did not permit the complete formalization of the transliteration.

We can state that only two letters (***ы*** and, especially, ***я***) in the Moldavian Cyrillic script represent ambiguous cases, and they have a small number of variants. Meanwhile the fixing of the proper selection in an unexpectedly complicated problem, even for native Romanian speakers. The first attempt used the statistical analysis of Romanian words. The developed descyrillization program works with approx. 0.15% of errors. Descyrillization can be used as a tool at investigation of historical development of the Romanian language, and for restoration of printed cultural heritage.

We plan the following development in this direction. For the software, a spelling checker would be integrated into the transliteration program. Other approaches to transliteration could be tried: dynamic contexts, syllable-to-syllable mapping, neural network. Word list in the Moldavian Cyrillic script is under preparation. It is also possible to cover the transitional Romanian alphabets (mixed Latin and Cyrillic), and 47-letter Romanian Cyrillic alphabets.

Acknowledgements

We are proud to acknowledge Dr. Hab. A. Alhazov for his help in collecting and interpreting the statistics and debugging the transliteration program.

References

E. Boian, C. Ciubotaru, S. Cojocaru, A. Colesnicov, L. Malahov (2014). Digitizarea, recunoaşterea şi conservarea patrimoniului cultural-istoric [Digitization, recognition and conservation of cultural and historical heritage.] *Akademos*, Nr. 1(32), 2014, pp. 61–68. – In Romanian.

M. Petic, D. Gîfu (2014). Transliteration and Alignment of Parallel Texts from Cyrillic to Latin, In: *Proceedings of LREC-2014*, Reykjavik, Iceland, 26–31 May 2014, pp. 1819–1823.

V. Demidova (2014). Particular Aspects of the Cyrillization Problem. In: Proceedings of the Third Conference of Matematical Society of the Republic of Moldova, Chişinău, 19-23 August, 2014, pp. 493–498.

Acvariu (1988): В.А. Уфнаровский. Аквариу математик. Кишинэу, «Штиинца».

ELRR: http://www.math.md/elrr/res_main.php

Dexonline: Dicţionar explicativ a limbii Romane: https://dexonline.ro/

Frequency: List of Romanian words by frequency: https://s3.amazonaws.com/101languages/common-words/romanian.xlsx

DESCRIPTION OF THE ROMANIAN SYNTAX WITHIN UNIVERSAL DEPENDENCY PROJECT

VERGINICA BARBU MITITELU, ELENA IRIMIA

Research Institute for Artificial Intelligence "Mihai Drăgănescu", Romanian Academy
{vergi, elena}@racai.ro

Abstract

This paper presents the effort of describing the Romanian syntax within the Universal Dependency project. This means, on the one hand, adopting some principles of syntactic annotation that have never been used for Romanian. On the other hand, it implies either using already established relations for the description of linguistic phenomena (the set of universal relations) or postulating new relations (language(s) specific), subsumed to the universal set. Mapping the existent Romanian treebanks to this annotation style is not at all trivial and manual intervention is required.

Keywords: universal dependency, Romanian, syntactic relations.

1. Introduction

Given the existence of two Romanian treebanks already (UAIC-RoTb – Perez, 2014; Mărănduc and Perez, 2015, and RACAI-RoTb – Irimia and Barbu Mititelu, 2015), each using a partially different set of syntactic labels, although sharing the same annotation principles, we consider that it is high time we offered a uniform grammatical analysis of the sentences they contain. What is more, we aim at an annotation methodology in line with international practices. At the same time, we envisage using the resulted treebank (which we call RoRefTrees) for training a (semantics-aware) (Romanian) parser.

This paper introduces the dependency relation set developed for Romanian in line with the Universal Dependencies[1] (UD) specifications. The work was done as part of the task of mapping the two different Romanian treebanks into the UD format, the results of which have already been released (16th November 2015) partially on the UD platform (http://universaldependencies.github.io/docs/#language-u).

UD is a project that aims at unifying treebank annotation cross-linguistically. Such a standardization initiative will encourage the use of syntactic information into multilingual complex applications, specifically in those involving Machine Translation (MT), which is the key domain promoting and sustaining multilingualism. A unified framework for developing treebanks can also be beneficial for cross-lingual learning and research. It is therefore very advantageous to align the Romanian treebanks to the UD annotation scheme, bringing it into an

[1] universaldependencies.github.io/docs/introduction.html

international initiative that gathers, at this point, 33 languages (in 37 treebanks) from all over the world.

The UD syntactic annotation scheme grew from the Stanford type dependencies for English (de Marneffe *et al.*, 2006; 2008; 2014) and aims at becoming a universal inventory of dependency relations (together with principles and guidelines for annotation). To preserve language specificities, the UD standard allows for extensions but in a consistent manner: if for a specific language the need arises to refine the inventory so that to accommodate some important particular phenomena, first denominations and solutions already used for other languages that display the same phenomena are checked, so as to avoid renaming or other types of incongruences in the inventory. Only when none of the languages in the project displays a certain phenomenon or no solution has been proposed for it so far, can a new relation be coined. And this new relation is a subtype of a relation already existing in the UD set (and, implicitly, in the respective language).

2. Dependency grammars

The dependency grammar (Tesnière, 1959; Mel'cuk, 1987) is the formalism chosen by both UAIC-RoTb and RACAI-RoTb for syntactic analysis, due to its characteristics:

- minimal structures: each node of the structure is a word in the analysed sentence; thus, no artificial nodes are present in the structure, not even gaps;
- ordered structure: the order of the nodes reflects the order of the words in the sentence;
- multiple branched structure: the root of the sentence is the main verb and all its dependents, irrespective of their number, are attached to it.

Moreover, given the relatively free word order of Romanian, the dependency formalism is better suited for its syntactic analysis. Furthermore, the transparent encoding of predicate-argument structure makes the dependency-based syntactic representation useful in MT and information extraction (Ding and Palmer, 2004; Quirk *et al.*, 2005; Culotta and Sorensen, 2004).

3. General annotation principles in UD

3.1 Each word has a head

Like any variation of the Dependency Grammar formalism, UD is based on an inventory of dependency relation types that can hold between two words in a sentence. The two terms in a dependency relation are called *head* and *dependent*. Each word in a sentence is a dependent of a single other word. The only exception is the word functioning as root: it is a dependent (via the `root` relation) of an artificial construction, the ROOT of the sentence. A head can have more than one dependent.

As a consequence, the dependency analysis for the whole sentence is a specific type of acyclic graph: a tree.

3.2 Heads are content words

To achieve a maximal amount of parallelism across languages, UD centres its analysis on *content words*, which are always heads of the dependencies, while *function words* (prepositions, conjunctions, auxiliaries, including modals) attach as dependents to content words and *punctuation* to the head of the clause or phrase that contains it. The content words are the ones that introduce less amount of variation between languages; therefore, it makes sense to focus on them if we want to produce a backbone of a syntactic tree for a specific sentence that can be easily translated into other languages. Conversely, function words are used in some languages to express, for instance, grammatical categories (e.g., case) that are expressed in other languages by morphological means (inflection).

Normally, function words do not take dependents and stand as siblings if more function words depend on the same content word. However, there are exceptions to this rule, in the treating of:

- multiword function words: fixed expressions, whose words are connected in a flat way to the first word, which is considered the head of the expression;
- coordinated function words (see below the treatment of coordination);
- function word modifiers (like negation or light adverbials);
- promotion by head elision: a function word can be promoted to the function of the missing content word (see below the treatment of ellipsis).

3.3 Clausal and non-clausal dependents

An important distinction made within UD is that between clausal and non-clausal dependents of a head. In spite of the fact that either realisation can fill in the same verb valency, they are labelled differently: e.g., the non-clausal direct object is labelled as `dobj`, while its clausal realization is `ccomp`.

3.4 Finite and non-finite clausal dependents

The classification of the *clausal dependents* does not differentiate between finite and non-finite clauses, but is instead based on the following distinctions: core vs. non-core arguments, subjects vs. complements, subjects of passives vs. other subjects, clauses with obligatory control vs. clauses with other types of subject, attachment to predicates vs. attachment to entities.

We want to highlight the fact that non-finite verb forms are treated as heads of subordinated clauses, thus allowing for a uniform treatment of the verbal arguments irrespective of its finiteness or non-finiteness.

3.5 Copular verbs

Copular verbs are treated, at least in the current version of the UD guidelines, inconsistently. On the one hand, the copula "to be" (in Romanian "a fi") is not the root of a sentence (the motivation behind this being that there are languages that use no verb to render the meaning of constructions with copula "be"); the root is the complement of the copula "be" (see Figure 1). Other copular verbs are heads of the clauses containing them and their complements are a special type of complements (see Figure 2), called `xcomp`.

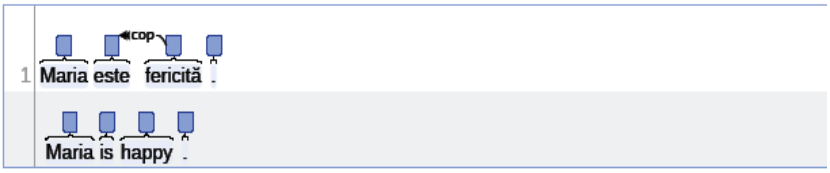


Figure 1. Treatment of the copular "fi".

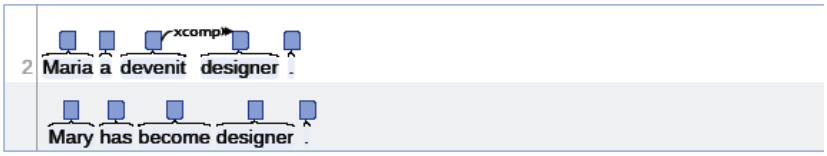


Figure 2. Treatment of other copula verbs.

On the other hand, in spite of the decision of not treating the copula "be" as a head, there is one exception to this: when its complement is a clause. In this case the copula is the head and the subordinate clause is in *ccomp* relation with it:

Figure 3. Copula verb "be" as ROOT.

3.6 Coordination

Coordination is treated asymmetrically: the first conjunct is the head and the others (as well as the coordinating conjunction) are attached to it as dependents (Fig. 4). When a dependent is shared among conjuncts, it is attached to the coordination head.

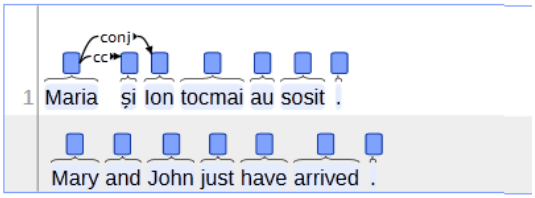


Figure 4. Coordinated elements.

3.7 Ellipsis

Ellipsis is one of the phenomena for which UD still lacks a consistent, not problematic analysis. Thus, different teams in the UD project use different methods for marking it. For ellipsis within sentence boundaries, also involving coordination, thus with usually parallel structures within the same sentence, a good solution is the use of a special relation, `remnant`, linking the dependents in an elliptic structure to the corresponding words in the non-elliptic part (Fig. 5). For ellipsis without parallel structures, this relation proves useless and the solution would be the promotion of one of the left dependents of the elided head (see Fig. 6, where the verb in the main clause is elided and its complement becomes the root of the sentence). Stipulating an empty node in the structure contradicts the characteristics of dependency grammars (i.e., the minimal structure of analysis). For Romanian, we use the first two strategies. No empty nodes exist in the trees.

All these three ways of dealing with elliptical structures (the remnant relation, the promotion of the most prominent argument and postulating a gap in the tree) have their advantages and disadvantages. For now, no common treatment of this phenomenon is found in the languages in UD and will be agreed upon in further releases. The most important thing is, for the moment, the consistent annotation of ellipsis within the same treebank.

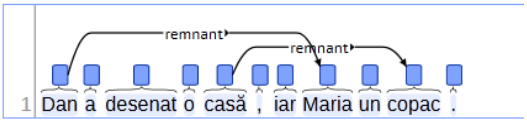


Figure 5. Ellipsis with parallel structures

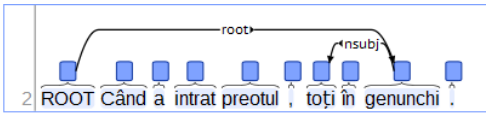


Figure 6. Ellipsis with promotion.

4. UD Relations for analysing Romanian

The relations used in the annotation of Romanian sentences are presented in Figure 7. Most of them are borrowed from UD, of course.

Core dependents of clausal predicates			**Non-core dependents of clausal predicates**			**Special clausal dependents**		
Nominal dep	*Predicate dep*		*Nominal dep*	*Predicate dep*	*Modifier word*	*Nominal dep*	*Auxiliary*	*Other*
nsubj	csubj		nmod	advcl	advmod	vocative	aux	mark
nsubjpass	csubjpass		***↳nmod:pmod***	***↳advcl:tcl***	***↳advmod:tmod***	discourse	auxpass	punct
dobj	ccomp	xcomp	***↳nmod:tmod***		neg	expl	cop	
iobj	***↳ccomp:pmod***		**↳nmod:agent**			**↳expl:pv**		
						↳expl:pass		
						↳expl:impers		
						↳expl:poss		
Noun dependents			**Compounding**			**Coordination**		
Nominal dep	*Predicate dep*	*Modifier word*	compound	mwe		conj	cc	punct
nummod	acl	amod	name	foreign	goeswith		**↳cc:preconj**	
appos		det						
nmod		neg						
Case-marking, prepositions, possessive			**Loose joining relations**			**Other**		
case			list	parataxis	remnant	*Sentence head*	*Unspecified dependenc*	
			dislocated		reparandum	root	dep	

Figure 7. Relations used in the annotation of the Romanian trees.

We distinguish among

- subjects realised as nominals (`nsubj`);
- subjects realised as clauses (`csubj`);
- subjects realised as nominals in passive constructions (`nsubjpass`);
- clausal subjects in passive constructions (`csubjpass`);
- direct objects (`dobj`);
- indirect objects (`iobj`). In general these two complements correspond to what is called direct, respectively indirect object in Romanian linguistics. However, in order to offer an analysis of the secondary object (the second accusative complement of a verb) within the economy of the relations labels, using the semantic roles of the two accusatives (the animate one is the addressee, and the non-animate one is the patient), the same two relations are used for their analysis: thus, the animate object is the `iobj`, while the non-animate is the `dobj`;
- the clausal realisation of direct and indirect objects (`ccomp`);
- the relation `xcomp` is used for many types of syntactic structures: all kinds of secondary predicatives and complement clauses with controlled subjects;

- the relation `expl` is used for the doubling phenomenon (for direct object, indirect object and subject doubling alike) (see Fig. 8 for an example with a doubled direct object), as well as for non-referential pronouns (Fig. 9) and expletive negation (Fig. 10).

Figure 8. Analysis of a doubling clitic.

Figure 9. Analysis of a non-referential pronoun.

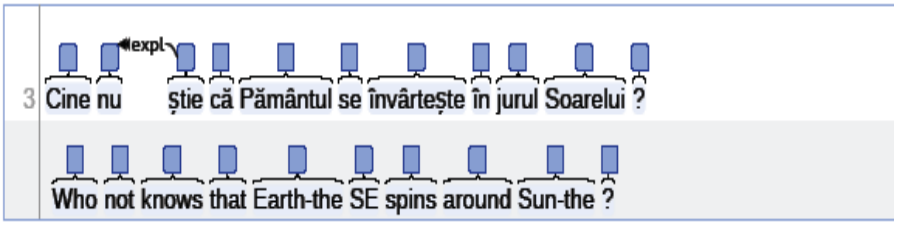


Figure 10. Analysis of expletive negation.

Non-core dependents of predicates and modifiers of nouns establish with their head relations that are named after the part of speech realising them:

- `advmod` (realised as adverbs),
- `nmod` (realised as (pro)nouns),
- `advcl` (adjuncts realised as a clause),
- `nummod` (realised as numerals),
- `amod` (realised as adjectives),
- `acl` (noun modifier realised as a clause),
- `det` (all determiners, be they articles, possessives, quantifiers, etc. are labelled as such).

`Case` is the relation linking the preposition (or postposition) to its head in Romanian. `Mark` is used for subordinating conjunctions.

We will skip the description of the other relations as they carry rather morphological (aux and auxpass, vocative, compound) and discourse interpretation (list, reparandum, foreign, parataxis, goeswith, dislocated), or mark the punctuation (punct).

In Figure 7, the boldfaced relation labels (i.e., nmod:agent, expl:pv, expl:pass, expl:impers, cc:preconj) are borrowed from other languages in UD displaying the same linguistic phenomenon. The arrows in the table are used to mark that the newly coined relations are subtypes of the UD ones. The agent in a passive structure is linked by the relation nmod:agent to its head. All subtypes of the expl relation are used for the various meanings of clitics: expl:pv is for clitics occuring with pronominal verbs, expl:pass for clitics in constructions with passive meaning, expl:impers for impersonal clitics. The initial conjunction in correlative constructions (e.g., *sau... sau...* "either…or…") is analysed as cc:preconj.

5. Romanian specific relations

In this section we will focus on the characteristics of Romanian that cannot be captured by the UD relations, thus being necessary to stipulate new subtypes of them.

The relation labels that are both boldfaced and italicised in Figure 7 above (i.e., ccomp:pmod, nmod:pmod, nmod:tmod, advcl:tcl, advmod:tmod, expl:poss) are Romanian-specific in the sense that they do not necessarily coin relations non-existent in other languages, but relations for which no special relation label has been provided so far within the UD community.

We chose to mark in a distinct way all occurrences of prepositional objects that are complements, not adjuncts, i.e. obligatory valences of the predicates taking them. Thus, we proposed the label nmod:pmod, which attaches the nominal in such a phrase to its head, while the preposition is analysed as case for this nominal. When this valence is syntactically realised by a (finite or non-finite) clause, its head is linked by the relation ccomp:pmod to its predicate.

For the moment, the labels nmod:tmod (time adverbial realised as a noun or prepositional phrase), advcl:tcl (time adverbial realised as a clause), advmod:tmod (time adverbial realised as an adverb) are used only for the trees originating in UAIC-RoTb, because such semantic information was annotated in this treebank. However, we aim at adding this label to pertinent cases in trees originating in RACAI-RoTb, as well.

The Dative clitic expressing possession is linked by the relation expl:poss to its verbal head: see Figure 11.

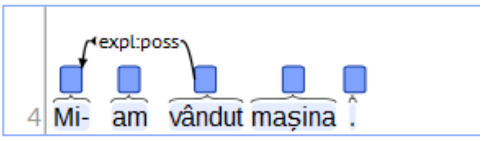


Figure 11. Possessive clitics

6. First release of Romanian data in UD

In November 2015 a first part of RoRefTrees (633 trees with 12094 words) was released within the UD project. The mapping from the annotation style of UAIC-RoTb and RACAI-RoTb was done at first fully manually, then automatically, training the dependency parser MaltParser on the manually UD-annotated sentences and using the learned statistical model to annotate the remaining trees. The resulting analysis is not a very exact one (57% LAS score, since the training corpus was very small: 300 trees), but we used it to learn correspondences between it and the original non-UD analysis in each of the treebanks.

A mapping algorithm was developed, following the steps:

1. For each pair of words identically linked in the two (UD and non-UD) analyses, count the frequency of the pairs of relations: (non-UD, UD) and (UD, non-UD);
2. Sort the mappings (correspondences between non-UD and UD) in decreasing order of their frequency. If (A,B), the most frequent mapping of type (non-UD, UD), is followed in the frequency order by (B,A) which is a (UD, non-UD) type, map A (non-UD) to B (UD). Traverse the correspondences lists and eliminate all the pair of this type;
3. Map all the remaining non-UD relation to the first two most frequent UD relations;
4. Use the mapping table dearived in 1.1-1.3 and, without altering the syntactic structure, replace in the corpus the non-UD labels with the corresponding UD ones.

The automated mapping was then inspected and manually corrected: the experience of correcting the automatic mappings instead of manually mapping from the scratch showed important reduction in terms of time and effort.

7. Conclusions and future work

For RoRefTrees we target a treebank of at least 9500 trees, annotated according to the UD standards, which will be released in May 2016, i.e. in the next UD release. They will be used for training a parser for Romanian, which will be made semantics-aware, with the hope of improving results over available parsers.

Acknowledgements

This work was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS – UEFISCDI, project number PN-II-RU-TE-2014-4-1362.

References

Culotta, A., Sorensen, J. (2004). Dependency tree kernels for relation extraction, in *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL'04)*, Barcelona, pp. 423–429.

de Marneffe, M.-C., MacCartney, B, and Manning, C.D. (2006). Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC.*

de Marneffe, M.-C., and Manning, C.D. (2008). The Stanford typed dependencies representation. In *COLING Workshop on Cross-framework and Cross-domain Parser Evaluation.*

de Marneffe, M.-C., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., and Manning, C.D. (2014). Universal Stanford Dependencies: A cross-linguistic typology. In *Proceedings of LREC*.

Ding, Y., Palmer, M. (2004). Synchronous dependency insertion grammars: a grammar formalism for syntax based statistical MT. In *Proceedings of the Workshop on Recent Advances in Dependency Grammar*, Geneva, pp. 90–97.

Irimia, E., Barbu Mititelu, V. (2015). RACAI-RoTb: nucleu de corpus de limbă română adnotat sintactic cu relaţii de dependenţă, *Revista Română de Interacţiune Om-Calculator* 8 (2), pp. 101-120.

Mărănduc, C., Perez, C.-A. (2015). A Romanian dependency treebank, CICLing 2015, Cairo, 14-20 April.

Mel′cuk, I. A. (1987). Dependency syntax: theory and practice, Albany, State University Press of New York.

Quirk, C., Menezes, A., Cherry, C. (2005). Dependency treelet translation: syntactically informed phrasal SMT. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, Ann Arbor, MI, pp. 271–279.

Perez, A.-C. (2014). Resurse lingvistice pentru prelucrarea limbajului natural. PhD thesis, "Al. I. Cuza" University, Iasi.

Tesniere, L. (1959). Éléments de syntaxe structurale, Paris, Klincksieck.

ONLINE LANGUAGE ANALYSIS: FACEBOOK VS. RESEARCHGATE

MIRELA TEODORESCU

Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iaşi

mirela.teodorescu@info.uaic.ro

Abstract

This paper proposes the analysis method of on line language, focused on comparing two communities, Facebook and Research Gate. Our goal is to highlight the language differences of social networks types, advantages (instant messaging, wide friendship, spread information) and disadvantages (not consistent and malicious comments, misinformation, disputed factual claims, veracity, temporal validity, etc.), what type of network suits the individual profile for different purposes (e.g. fun), for academic individuals, for specialized services (marketing, advertising, etc.), and so on. The study may be of interest to linguists, sociologists, PR firms, businesses investors, and, public, in general, in order to make a relevant selection of their purposes.

Keywords: Facebook, ResearchGate, discourse, language analysis, linguistic features.

1. Introduction

Social network is a synergy of interdisciplinary communication (anthropology, sociology, history, social psychology, political science, human geography, biology, economics, communications science, sciences and other disciplines who share an interest in the study of the empirical structure of social relations and associations that may be expressed in network form) providing a place where ideas are confronted, updated, spread, shared, debated, etc. On the social networks "users" shares their real thoughts, they are not censored (because of the secured identity), free thinking on one hand, but also the comments can be without value or attack to somebody, or everything else. This variety of messages could be interesting for a natural language analysis.

The study is focused on the impact of Social Networks (here, *Facebook* vs. *ResearchGate*), in our life, how they influence our thinking, interacting, learning, consolidation relationships, understanding of the society. This content is an important source for NLP (*Natural Language Processing*) area, source for discourse analysis on this side of communication. The aim of this study is to analyse the contents of supplied data of social networks from different categories of publics, suggesting ways of their identification and then to clarify the descriptive consumer in dialog behaviour influenced by the amount of messages according to their status and purpose, or their education.

In this regard, we want to emphasize as novelty the diversification of messages, domains, social actors (individuals and/or organizations) interacting in communication process, spontaneously and sincerely, also the different approach of messages in different social networks.

The study is structured as follows: after a short introduction about the importance of this topic, the section 2 reveals discourse analysis characteristics in linguistics; the section 3 presents the language features in selected social networks; section 4 describes a case study of manual annotation of texts, results and interpretation and finally, the section 5 presents the conclusions and future work.

2. Background

Discourse is the result of creation and organization of the segments of a language, through the sentence. The segments of language can be bigger or smaller than a single sentence but the adduced meaning is always beyond the sentence. Discourse is "any coherent succession of sentences, spoken or written" (Matthews, 2005). The links between sentences in connected discourse are as much important as the links between clauses in a sentence. From formalism assumptions the discourse is "language above the sentence or above the clause" (Stubbs, 1983). Michael Stubbs says that, "any study which is not dealing with (a) single sentences, (b) contrived by the linguist, (c) out of context, may be called discourse analysis" (Stubbs, 1983).

In other words, from sentences in isolation to utterances in context: to study language in use is to study it as discourse. This is a fact that "knowledge of a language is more than knowledge of individual sentences". Spoken discourse is generally characterized by normal non-fluency, which refers to unintended repetitions, fillers, false starts, grammatical blends and unfinished sentences (Leech *et al.*, 1993). Written discourse, on the other hand, does not, naturally, face such phenomena and as a result it appears more fluent.

The discourse used on social media is a discourse of global sense. The global discourse topic is determined by contextual, cognitive and pragmatic factors such as the extra linguistic situation in which the communicative interaction takes place, the general knowledge that both the writer and reader share, the purpose of the text.

Through discourse analysis the user identifies and count relations between different parts of speech, to put in evidence patterns of use (Gîfu and Cristea, 2012).

Broadly construed, NLP is considered to involve at least the following subtopics: Phonetic and phonological knowledge, Morphological knowledge, Syntactic knowledge, Semantic knowledge, Pragmatic knowledge, Discourse knowledge, World knowledge (Allen, 1995).

3. Language features on social networks

Understanding the language in nowadays society, it is associated to understand the social networks in which language is embedded. A social network is a way of

describing a particular speech community in words and terms of relations between individual members in a community, the specific language, academic or colloquial.

According to various sources cited by Saunders and Goldenberg, "academic language refers to the specialized vocabulary, grammar, discourse/textual and functional skills associated with academic instruction and mastery of academic materials and tasks" repetition as linguistic features (Saunders and Goldenberg, 1999). Academic Language consists of a rigorous vocabulary and it is used in the scientific universe of discourse. In general, we talk about sophisticated vocabularies, sentences that start with "and" or "but" or with transition words, such as "however", "moreover" and "in addition", sentences in which slang is replaced with accurate descriptors, appropriate for use in all academic and work place settings (Dutro and Moran, 2003).

On the other hand, colloquial language consists of a variety of slang, vulgar language, informal words, or phrases in a piece of writing. We talk about two kinds of languages used in different context, formal and informal. Formal is associated with scientific language, and informal with the colloquial one.

> ***Informal is a broad term and colloquial falls in the definition of informal. Every colloquial term, sentence, or speech would be informal. Colloquial language or language of common use refers to conversation and not to written language.***

As textual features of language can be: alliteration, assonance, consonance, hyperbole, voice, metaphor, onomatopoeia, oxymoron, person, personification, repetition, sibilants, symbolism, tone, word choice, rhyme, rhythm, sound, parallel construction, triple construction, simile, use of slang, use of direct or indirect speech, use of incorrect grammar, pun, litotes, use of multiple adjective or adverbs, simple sentences, compound sentences, complex sentences, rhetorical question, use of command, use of first and second person pronoun, euphemism, neologism, listing, sarcasm, irony, contrast, use of numbers/statistics, use of authority figures, quotation.

These features can be common to academics and colloquial language, or can be specific. In this regard, we want to study and emphasize several features, and, especially, to find out which of them differentiate the selected social networks and how they influence the social media. Thus, emoticons are largely used by Facebook users to express emotional prosody. Emotional prosody is characterized as an individual's tone of voice in speech that is conveyed through changes in pitch, loudness, timbre, speech rate, and pauses, which is different than linguistic and semantic information (Erekson, 2010).

3.1. ResearchGate

ResearchGate, as a social network, was built by scientists and for scientists. It was founded in 2008 by physicians Dr. Ijad Maadisch and Dr. Sören Hofmayer, and

computer scientist Horst Fickenscher, today ResearchGate has more than 8 million members. ResearchGate mission is to connect researchers and to make easier for them to share and access scientific researches.

The New York Times described the ResearchGate site (Lin, 2012), as a mashup of Facebook, Twitter and LinkedIn. Many of its features are typical among social network sites, such as user profiles, messages that can be public or private, methods for finding other users with similar interests. But, ResearchGate differs from other social networks in that it is designed for researchers and scientists.

Conversation strings are focused on a research interest or paper and a user can "follow" a research interest. It has also a blogging feature for those users who want to write short reviews on peer-reviewed articles. ResearchGate indexes self-published information on user profiles and suggests members to connect with those who have similar interests. An option allows a user to post a question; this is forwarded to scientists identified on their user profile to have relevant expertise. ResearchGate, also, has private chat rooms where scientists can share data, edit shared documents, or discuss confidential topic.

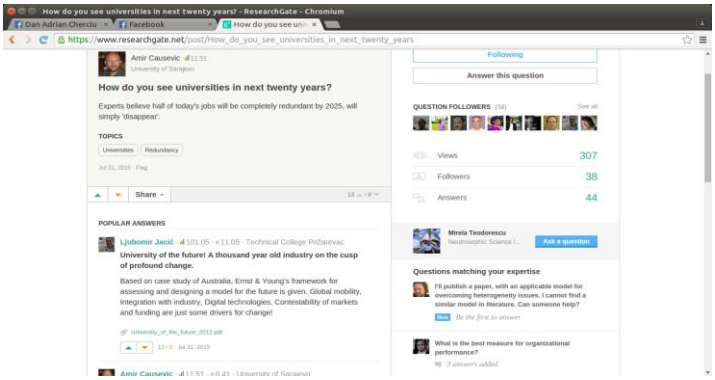


Figure 1. ResearchGate – Questioning option

According to users' activity, ResearchGate calculates a "RG score", a metric that measures scientific reputation based on how all of user's research is received by his/her peers (research papers, questions, answers).

3.2 Facebook

The social network Facebook, is an online social networking service known as the most popular one. All over the world, it has more than 60 million active members. The users can upload photos, have group discussions, and even play games on their individual profiles; they can also add one another as "friends" and connect with users who share similar interests, regardless of where they are in the world. Currently, more businesses and corporate folks are joining Facebook, posting their

pages to the Facebook network. Advertisers are even turning their attention to this growing market.

Facebook is a social network for all kind of individuals, for teenage, academic, friends, business companies for advertising, political and propaganda aims.

The like button represents a social networking feature, allowing users to express their appreciation “like” or “not like”, of content such as status updates, comments, photos, and advertisements.

We have to mention that the website has won awards such as placement into the "Top 100 Classic Websites" by PC Magazine in 2007, and winning the "People's Voice Award" from the Webby Awards in 2008.

Figure 2. Facebook – Posting area

4. Case study

To study the discourse of two social networks, we selected 50 posts from the mentioned social networks, ResearchGate and Facebook. We note that ResearchGate is a social network in which the communication language is English, the common language for all researchers. On the other hand, Facebook, is a huge social network; regarding communication language it is customized for each friends' groups language of each country; in our study we will deal with Romanian.

Further, to have a suggestive and objective metric, we will choose the appropriate language features which will exemplify the analysis both in English and Romanian.

Among other things, we decided to annotate features such as: emoticons; constructions involving *however*, *moreover*, *how*, *further*; repetitions; variety of words; onomatopoeia; slang; sophisticated vocabulary sentences; fluency; like button; words; sentences.

Annotation of the text

The annotation of the texts was made manually according to selected features of the language, each of them marked in a different colour. A sample of the annotation is shown in Figure 3.

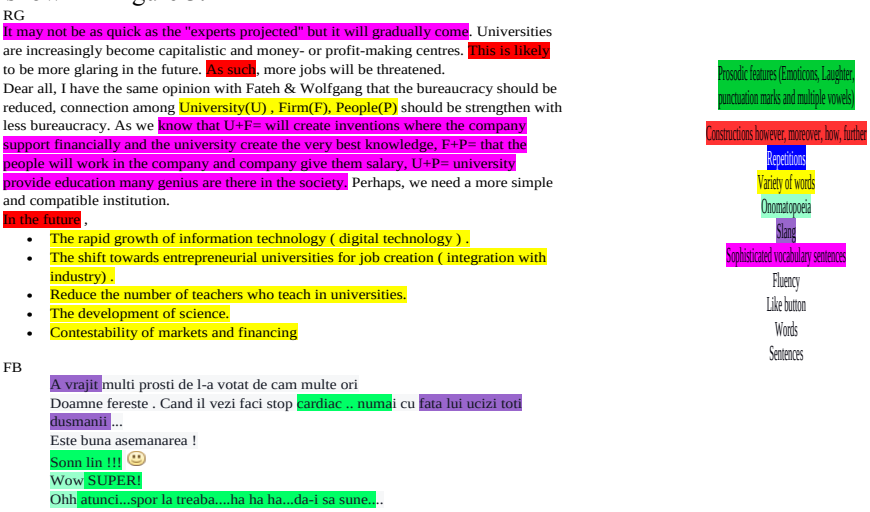


Figure 3. Sample of annotation of selected texts

ResearchGate has the option of posting questions and online answers. The question is addressed to all researchers belonging to a certain research domain. The operation is performed as a brainstorming, each researcher can answer, offering ideas, giving links for others' research studies, they can ascertain more or less concerning the suggested theme of debate (see **Figure 4**). The answers are qualified of high level and written in an academic language.

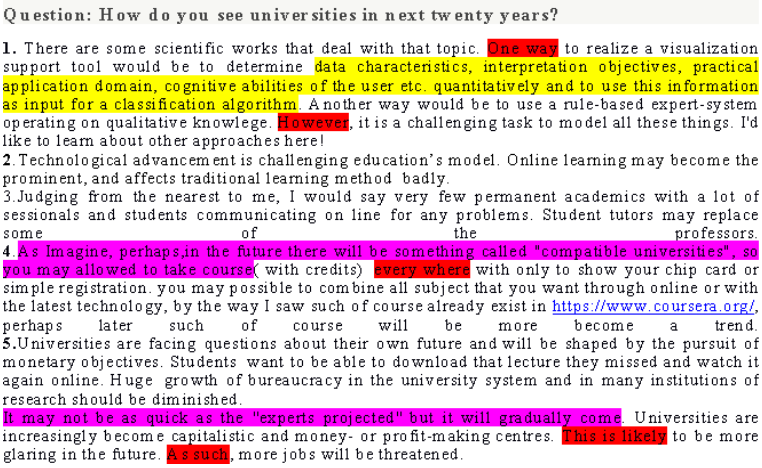


Figure 4. ResearchGate – A sample of brainstorming

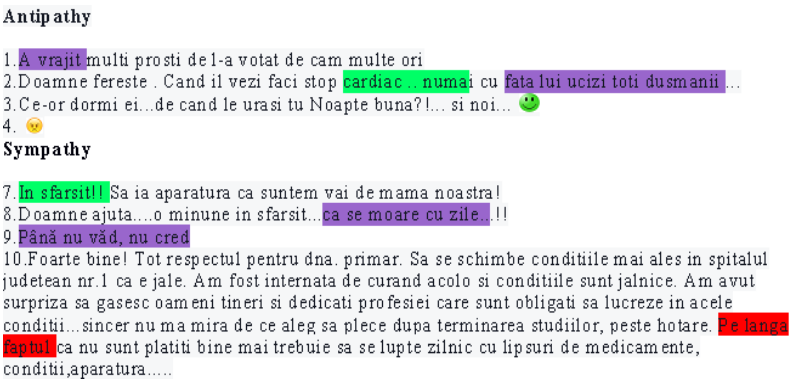


Figure 5. Facebook – Expression of antipathy/sympathy

Facebook, through its structure, allows users to post everything they want on their account, informing everybody related to their relationships, but also the friends of their friends. The language of comments on this social network is colloquial. These posts only have the purpose to inform users, not to offer solutions to problems. There are only few academic language comments. When a user does not like a post, she/he can answer malicious comments, misinformation, disputed factual claims, with veracity, or temporal validity (see Figure 5). In this manner, permanently, challenges among users arise, which show sympathy or antipathy for individuals.

To perform this analysis, we made a manual annotation and we selected the following features: constructions *however*, *moreover*, *how*, *further*; variety of words; sophisticated vocabulary sentences; fluency; words; sentences; prosodic features (emoticons, laughter, punctuation marks, and multiple vowels); slang; like button; onomatopoeia (see Figure 6).

		RG	FB
1	Prosodics features	1	27
2	Constructions however, moreover,...	57	4
3	Repetitions	0	1
4	Variety of words	19	1
5	Onomatopoeia	0	10
6	Slang	0	8
7	Sophisticated vocabulary sentences	50	4
8	Fluency	50	6
9	Like button(hundreds)	2,53	4,21
10	Words(hundreds)	61,75	8,53
11	Sentences(hundreds)	4,33	1,1

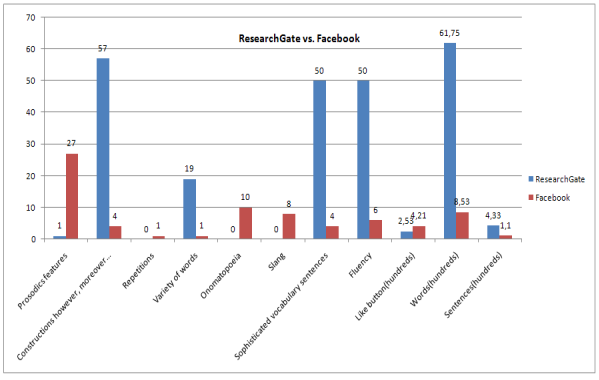


Figure 6. Statistics on the manual annotation of the selected texts

The language features such as: constructions *however*, *moreover*, *how*, *further*, variety of words, sophisticated vocabulary sentences, fluency, words, and sentences

are proper to the academic language used by researchers on ResearchGate social network. The language features such as: prosodic features (emoticons, laughter, punctuation marks, and multiple vowels), slang, like button, onomatopoeia are specific to the colloquial language widely used on Facebook social network. A graphical representation is also used to emphasize the differences between the academic and colloquial languages on social networks.

In case of the social network Facebook, there are also posts in which we found features specific to academic language, such as sophisticated vocabulary sentences, fluency, variety of words. This means that among users are also individuals who belongs to an academic education.

4. Results and interpretation

Table 1 shows a comparative analysis and interpretation of the features specific to the two types of social networks.

Table 1. Interpreting language features

Features	ResearchGate	Facebook
1. Prosodic features (emoticons, laughter, punctuation marks, and multiple vowels)		
	On academics language there are not figures of prosodic features.	Most of the posts consist of emoticons, also laughter, and multiple vowels. Through these prosodic elements the users state their status of joy, sadness, disgust, intrigue, and acceptance or disagreement.
2. Constructions with *however*, *moreover*, *how*, *further*	***However***, it is a challenging task to model all these things.	***Totuși*** ar fi, e tragic 
	Through this feature the fluency is maintained and ideas, concepts or ascertained methods are introduced or highlighted by the users. All posts of more than 5 sentences use such a feature.	This kind of feature is found only on few posts belonging to users who want to offer some explanations regarding a subject. They are triggered by a challenge, malicious comments, misinformation, disputed factual claims, veracity, temporal validity.
3. Repetitions		Când ***este, este***… 
	This feature is not found here.	It is used to emphasize an idea.

	to teach, to cooperate, research, promote novelty, to collaborate, to gather knowledge to select good reasons, to promote useful things, to save the planet.	
4. Variety of words	It is peculiar to academic language; it is an explanatory feature for academics language. It gives relevance to the explanation.	It is used only by a few users, giving some explanatory answers to posts representing challenge or misinformation.
5. Onomatopoeia		***Ohh***, atunci... spor la treaba... ha ha ha... da-i sa sune…
	It is not found here.	It is found to reveal the users' status, joy, happiness, sadness, acceptance, disagreement.
6. Slang		*Când îl vezi, faci stop cardiac... numai cu fața lui ucizi toți dușmanii…*
	It is not found here.	It is found on posts; there is a segment of users who often use this feature.
7. Sophisticated vocabulary sentences	*Becoming a change maker university which effectively leverages technology and human connectivity, open systems and access, and peer-to-peer and intergenerational learning will not be easy.*	
	It is used as a usual feature. All sentences are built according to this feature.	The users are writing in a hurry, it is no time for sophisticated vocabulary sentences. They use a simplified language.
8. Fluency	*Universities are facing questions about their own future and will be shaped by the pursuit of monetary objectives. Students want to be able to download that lecture they missed and watch it again online. Huge growth of bureaucracy in the university system and in many institutions of research should be diminished.*	

	It is an essential feature for academics language, and it generates continuity of ideas.	Only few posts contain fluency, those longer than 5 sentences.
9. Like button	It is used to emphasize the quality of an idea. Researchers give "like" for interesting subjects, ideas, debates.	It is used for agree with a user, subject, post, for everything, even because a friend of mine gave "like". The more "likes" you have, the more popular you are on the social network.
	The content is rich, the explanations are well documented. Thus the initiator of the question can find solutions for his research or for his subject.	There is a deficiency of words in the content of the posts. Most of them are emoticons.
10. Words	Each post is composed of many sentences for explanations, only few posts are simple.	Many times the sentences are truncated by prosodic features. The posts that consist of more sentences are explanations or argumentations of users for challenges, misinformation, claims, and veracity.

5. Conclusions and future work

Surely, this analysis is a pilot study, first of all our text is a small sample of messages from two social networks. But, we got some representative data regarding academic language used by ResearhGate "vs" colloquial language used by Facebook.

On ResearchGate the posts are only related to an initiated subject, while on Facebook there are many posts not consistent, some of them are malicious regarding content or to individuals who posted, some of them are subject of misinformation, manipulation, influencing, intoxication, propaganda, or disputed factual claims, veracity, temporal validity data, because Facebook can hide the real identity of the users.

On the other hand, social networks promotes information instantaneously (in case of Facebook) according to new technology, encourage wide friendship all over the world, solve some issues of researching domains.

This study containing a manual annotation is a beginning for the future research of social network language. We should like to extend this study to an automatic annotation of texts collected on social networks and to find also other distinctive features of analysis to give a more comprehensive image on different aspects of the social networks discourse.

References

Allen, J. (1995). Natural Language Understanding, second edition (Redwood City: Benjamin/Cummings).

Biber, D. (1988). Variation across speech and writing. New York: Cambridge University Press.

Biber, D. (1995). Dimensions of register variation: A cross-linguistic comparison. Cambridge: Cambridge University Press.

Dutro, S., Moran, C. (2003). Rethinking English language instruction: An architectural approach. In Garcia, G. (Ed.) English learners: *Reaching the highest level of English literacy*, pp. 227-258. Newark, DE: IRA.

Erekson, J. (2010). Prosody and Interpretation. In *A Journal of Literacy and Language Arts*, Vol. 50, Issue 2.

Gîfu, D. and Cristea, D. (2012). Public Text Categorization. In *Proceedings of The 8th International Conference "Linguistic Resources and Tools for processing of the Romanian language"*, Mihai Alex Moruz, Dan Cristea, Dan Tufiş, Adrian Iftene, Horia-Nicolai Teodorescu (eds.), "Alexandru Ioan Cuza" University Publishing House, Iaşi, pp. 75-84.

Leech, G, M., Deuchar & R. Hoogenraad. (1993). English grammar for today. London: Macmillan Press Ltd.

Lin, T. (2012). Cracking open the scientific process. In *The New York Times*. Retrieved 26 June 2014.

Matthew, D. 2005. Relationship between the order of object and verb and the order of adposition and noun phrase. In *The World Atlas of Language Structures, edited by Martin Haspelmath*, Matthew S. Dryer, David Gil, and Bernard Comrie. Oxford University Press.

Saunders, W., Goldenberg, C. (1999). The effects of instructional conversations and literature logs on limited and fluent English proficient students' story comprehension and thematic understanding. *The Elementary School Journal*, 99, pp. 277-301.

Stubbs, M. (1983). Discourse analysis: The sociolinguistic analysis of natural language. Cichago: Cichago University Press.

Wardhaugh, R. (2006), An introduction to sociolinguistics, 5th ed., USA: Wiley-Blackwell.

INDEX OF AUTHORS