UNIVERSITY "ALEXANDRU IOAN CUZA" OF IAŞI

FACULTY OF COMPUTER SCIENCE

Statistical and Inference Based Approaches in Natural Language Processing – Applications to Morphology and Syntax

- PhD thesis Summary-

Supervisor:

Prof. Dr. Dan Cristea, c.m.R.A.

PhD student:

Radu Simionescu

Iași

September 2016

This thesis presents the contributions resulted from various experiments, innovations, analysis and developments conducted in solitude or in collaboration with fellow NLP enthusiasts and researchers.

The inception of the doctoral research activity is driven by three developments. The first is the acquisition of a Romanian Treebank and the development of a statistical syntactic parser. Second is the construction of a noun phrase chunker. And the third is the improvement of the Romanian UAIC part of speech tagger (POS tagger), a tool for automatic morphologic annotation of Romanian text that has been developed in the project of my Master Degree (Simionescu, 2011c). The actions performed towards reaching these goals, have led to the development of Graphical Grammar Studio (GGS), which has captured the main focus of our activity, due to its impressive wide range of applicative use.

GGS is a tool that can now be used in many areas of NLP research, at different levels of language analysis. GGS is a symbolic system that can be used to analyse and annotate textual data in a user-friendly manner. Its entire workflow is mainly visual and it requires little understanding of artificial scripting languages, from its users. It now provides features that can facilitate many NLP tasks. Even though GGS has many features, which satisfy almost any NLP use case, it also provides a manner of combining its features with a scripting language to allow the extension of its basic functionality to serve less common NLP tasks.

The thesis highlights on the three main points of interest in NLP: resource development, automatic processing tools and applications. The thesis presents our contribution in all of these aspects of NLP.

In terms of resource development, the contributions bleed into various sub-fields of NLP, on different layers of analysis. On the morphologic layer, we present new methods to automatically expand the current morphologic dictionary of the UAIC Romanian POS tagger with diachronic entries, while conducting various experiments. In one of them we have successfully inferred a subset of the Romanian diachronic morphology. In another experiment we have trained a statistical model that captures the evolution of common character constructs of a lexicon.

Also in the resource development section, we have contributed in the bootstrapping process of the UAIC RoDepTreebank (Mărănduc & Perez, 2015). This process consisted in retraining a MaltParser model on a gold corpus with the purpose of extending it.

In terms of NLP tools and applications, we have contributed with various Romanian and multilingual tools.

The contributions to the UAIC RoDepTreebank also fructified in a Romanian dependency parser, based on the MaltParser framework. With this occasion, we also experimented with developing an in-house syntactic parser for Romanian based on a statistical model.

After experimenting with various machine learning and rules systems that are popular in NLP, the main focus of the doctoral activity has moved towards symbolic systems. In our experience, statistical models are easy to obtain once a sufficient amount of training data is available. No matter how precise, these systems are never 100% accurate in their predictions. By observing the errors that these systems produce, linguists usually feel the need to somehow control their behaviour in certain linguistic situations. This can be achieved only in hybrid systems, which mix statistical models with a manner of manually controlling the process. Rules systems are usually adopted to restrict, extend or override the behaviour of a statistical based system. Such

rules systems can also be used to develop NLP tools on their own, without the help of a statistical approach. There are various rules based system available for NLP applications. After experimenting with various examples, we have developed our own.

We designed and developed GGS, a framework for building rule based systems suitable for many NLP tasks. This represents the central focus of the thesis. GGS is a versatile tool. Its best description is: A NLP tool for finding and annotating sequences of tokens. A GGS expression can be used to query an input document and is defined in a graphical manner by a network of nodes, which describes the conditions that the matched sequences must fulfil. There are multiple types of conditions that can be imposed on the tokens of the sequences. They can refer to the attributes of the searched tokens, to the positions of the tokens inside the searched sequence or to relations with other tokens.

GGS has evolved along with the various tools that have been developed or enhanced with it. First, we have built a deep noun phrase chunker for Romanian. This has helped in identifying the limitations of GGS, limitations that would be overcome with future developments.

We improved the rules system of the UAIC POS-tagger, which is a hybrid system. We adopted a constraints system, based on GGS patterns.

We modelled a GGS verbal group chunker.

We experimented with a system that infers a GGS grammar, based on examples of sequences that the inferred grammar must find in a corpus.

We adapted GGS to work with big data, by employing a searching algorithm that makes intensive use of indexing techniques.

We researched the linguistic challenges introduced by social media communication style.

With the development of new features, GGS has gained more and more expressive power. A GGS pattern can exploit the various relations between tokens or between nodes of information in general. More precisely, GGS patterns can traverse nodes of information, based on the sequential relation, parent-child relation, or reference relation, to define valid structures, by imposing constraints of the attributes of the nodes. To achieve this efficiently, it makes use of an indexed search mechanism, which is specialized in, but not limited to, sequence finding.

The conclusions of the thesis emphasize the potential of GGS as a highly flexible framework for developing NLP tools and for conducting various linguistic analysis, due to its capability to search efficiently through structured data, especially because it allows users to define and organize parsing logic by using visual schematics.

Many of the innovations and developments presented in this thesis are the result of collaboration and coordination with other researchers and groups. This section underlines the personal contributions of the author, in the presented activities.

The implementation of the GGS Engine, Editor and AnnotationExplorer was performed entirely by the author. The manner in which patterns may be described by the user, including the adoption of network representation for RTNs and the syntax of the token matching expressions, are highly inspired by NooJ. Yet, GGS offers, on top of that, a set of features that stands out as novelty in terms of usability, functionality and in the manner in which they overcome implementation challenges. Most important among these, are:

- The JavaScript integration, which was primarily designed and developed to enable the use of variables, and which, due to its flexibility, can achieve much more than that. It allows the use of complex data structures. It empowers the user with a versatile scripting language that adheres to both object oriented and functional programming. The implementation of the JavaScript integration was challenging from multiple aspects. When backtracking, the JavaScript context must be reversed. This is achieved by maintaining stacks of clones of the entire JavaScript context. Cloning in itself represents another challenge, due to the functional nature of the language;
- Look-Ahead and Look-Behind assertions, which are inspired from regular expressions. They allow peeking on the stream of input tokens, which enables the possibility to explore the neighbouring context of a token, without consuming that context. The Look-Behind assertions bear an important implementation challenge, because they traverse the network backwards. The pattern used for Look-Behind assertions is not required to be fixed length, as is the case of regular expressions. This is achieved by compiling a mirrored version of the network and feeding it tokens in reverse order;
- Cross-reference assertions enable traversing tokens based on relations defined using reference IDs. Moreover, they allow the invocation of an additional search in a corpus, and joining the result.
- The indexed search capability allows the use of GGS as a corpus query tool for large volumes of data. It adopts a novelty solution regarding its use of bitsets to efficiently reduce the search space for a given pattern.

Interesting results have been obtained in the field of Grammar Inference. The similarity between a simple NP chunking grammar, used as an inducing grammar, and the induced grammar is fascinating. The presented state merging algorithm was also implemented entirely by the author. State merging algorithms are not new, but adapting them to network represented state machines and organizing the layout of the result into an easy to interpret network of nodes, is. This in itself represents a separate challenge that was overcome quite successfully using a three step process, which combines aspects of Graph Theory with Formal Language Theory:

1. a FSA minimisation step, which is performed by an algorithm specifically adapted to state machines represented as networks;
2. a reduction of intersecting arcs, which is obtained by transforming every directed, maximal, complete, bipartite sub-graph that has all the parents in a partition and all the children in the other, into a planar graph. This is achieved by inserting empty nodes in the network, which play the role of hubs. This transformation doesn't alter the behaviour of the network, while making it more legible;
3. a step dealing with positioning the nodes. This is achieved by laying out the nodes on a grid and determining their position based on a depth-first pass.

In the area of linguistic tools and resources, important contributions have been made, especially with respect to the Romanian language. The integration of GGS with the UAIC POS tagger, which has enabled the development of finer POS disambiguation rules, represents another personal contribution of the author.

In the area of diachronic studies, the author has been involved in the design, in the testing of the experimental methodologies and in the execution of two experiments that both proved successful methods of recovering obsolete morphologic entries of languages, given the appropriate resources. The experiment that deals with inferring obsolete entries that contain a difference in the stem of the word compared to their contemporary form, also involved a human validation, which was performed by a group of 20 students. The author has led the validation

process by providing documentation and support, and by constantly ensuring a high quality of the results.

Many of the novelty features of GGS have been designed to ease the development of a complex Romanian NP chunker, which has fructified as another linguistic tool, publicly exposed as a web application and service.

By collaborating with linguists, the author has contributed to the development of the UAIC RoDepTreebank and to the development of a Dependency Parser trained on it, based on the MaltParser tool. Like the POS tagger and the NP chunker, the parser has also been exposed publicly as a web application and service – an effort that is also attributed to the work of the author.

Another contribution is represented by the support offered by the author in experiments regarding the impact of non-standardized textual communication style, i.e. diachronic texts and modern chat, in training data for POS tagging and dependency parsing.

The Romanian tools that are publicly available as web applications and services, the UAIC POS tagger, the UAIC NP chunker and the UAIC Dependency parser, have been integrated as UIMA components by the author, as a task in the collaborative effort project – METANET.

The author is also actively involved in the SSPR project that aims to develop a Romanian treebank in the UD formalism and a syntactic parser that takes semantic information into account.

There are many improvements that are on the roadmap of future developments for GGS. One of the most important one is to allow patterns to access multiple resources while finding matching sequences. This will enable making cross-reference assertions in multiple corpora, and also will enable the use of external resources, such as databases, ontologies, dictionaries, thesaurus, etc. As a result, GGS will become capable of performing even more advanced matching tasks. A challenging aspect in this development is represented by the lack of an interface towards the internal representations of various resources.

The GGS matching process can use information that is structured as networks of hierarchical nodes. But it is currently limited regarding the manner in which such information may be stored. For NLP, on the syntactic layer, we have mainly focused on a common use case, i.e. the atomic node of information is the token/word. We offer the GGS user access to such data through the XML format, and we provide the means to generate an indexed version of that, to enhance the performance on frequently queried data. But linguistic information may be represented in a wide variety of formats. A future development of GGS, which currently represents a high priority, is to increase the amount of formats and information endpoint types that can be accessed from a query. Currently, only XML data may be processed from a local disk or local memory. To enable access to various existing external resources, it is required to create a bridge towards their internal representation formats. Some resources that are of interest in NLP applications are accessible as external data, such as databases or custom web service endpoints. Creating a bridge from GGS queries to their custom manner of interrogation is desired. One may extract information locally as XML and feed that to GGS, but that would be just a snapshot of a continuously improving resource and thus, should be avoided.

The current manner in which GGS accesses input data, must be decoupled as a customizable bridge, to enable integration with other types of information portals as well. A default set of bridges will be available in GGS, besides the current ones, which are composed of the XML

format and of a custom indexing scheme based on Lucene and provided by GGS. The XML format provides already a lot of flexibility and may enable accessing many external resources that are already available as XML. The next step is to enable access from GGS patterns to resources encoded in the CONLL format, or accessible by using various query languages such as CorpusQueryLanguage (CQL), KoralQL (Joachim & Nils, 2015), XPath or other database query languages. The next milestones to be reached in this direction are to enable users to access WordNet, FrameNet and eDTLR.

Users would also require tools to build their own custom GGS bridges with little effort. We explored strategies that would make such a process easier (Simionescu & Cristea, 2012), first by automatically inferring what type of information is present in previously unseen data. This ambition alone hides a lot of challenges which have not yet been addressed.

Bridging into other resource endpoints may not necessarily be limited to static information portals. A user may decide that a particular branch of the logic involved in taking a decision should better be solved by some other linguistic tool, such as a neural network or other custom algorithm. Bridging will allow integrating other tools specialized on particular linguistic tasks. GGS logic may be used to collect and prepare the input for such tools. GGS might even become usable as an integration tool, which only describes the flow of data through the various linguistic components of a complex system.

The field of Grammar Inference looks much more appealing from the perspective of recursive transitional networks applied for natural language. We have just managed to scratch the surface with a very promising prototype. Many challenges remain unexplored and are the subject of future research in this area. The most obvious task is discarding all the irrelevant information that is not necessary for determining when a particular syntactical phenomenon should occur. How would such an approach perform in a context with a lot of irrelevant information, or how would it perform in a situation where much of the available information is present in an external resource, such as an ontology, dictionary or thesaurus? How would generalization be achieved automatically?

Another important research opportunity in the field of GGS Grammar Inference is exploring the impact of assisted learning, by providing feedback to the learning mechanism for the assumptions that have a low trust score.

Moreover, we have only explored the idea of inferring GGS patterns based on the adjacency relations between tokens, using a state merging approach. In the context of information structured as networks of hierarchical nodes, GGS Grammar Inference could provide more than just generating patterns that match sequences of tokens. GGS patterns are not useful only for finding sequences of information nodes. The following use case of Grammar Inference reminds of the complexity that the human mind is capable to grasp effortlessly: A GGS pattern may be designed to find a single token that has a particular role in a complex relational network of nodes of information (tokens and constituents). Provided with enough positive and negative examples of such tokens, a learning process that explores not only the adjacency relation between nodes of information should determine that the features that separate the positive examples from the negative are contained not in the attributes of the provided tokens, but in the attributes of the tokens that they are related to, directly or indirectly.

The UAIC POS tagger has been integrated with a rule-based system that allows manual intervention in the behaviour of the disambiguation process. The rules have been designed manually, based on the results of error analysis. An interesting research opportunity related to Grammar Inference is to automatically generate the GGS patterns that find the contexts in

which a particular type of error occurs. These may be used to automatically create rules that correct errors of the statistically based disambiguation process.

The developed VP chunker adopts a strategy of detecting verbal phrases without imposing a rigorous set of constraints on the inner structure of such sequences of tokens. The VP is detected simply by including some words that are near a main verb, but the order and combinations in which such words are allowed are not described by the VP chunker. From this point of view, the VP chunking grammar consumes VPs successfully but it may generate incorrect sequences, from a linguistic point of view. An interesting research regarding Grammar Inference that would result in a more linguistically inclined VP chunker is suggested in Error! Reference source not found. as an automatic method of fine-tuning GGS patterns. The current VP chunker could be used as an inducing grammar to extract verbal phrases from a corpus and feed them as positive examples to an inference process. This would generate a GGS network that exposes the constraints regarding the allowed order of the components that are part of the Romanian verbal group. The various generated paths of GGS nodes would probably correspond to various types of Romanian verbal groups. With some adjustments the generated GGS network could be manually altered to also classify the matched sequences.

The methodologies of recovering morphological dictionary entries of obsolete word forms, proved efficient, and are convenient to perform because both are based on semiautomatic processes that make use of already existing Romanian linguistic resources. However, to obtain qualitative NLP results on old texts requires more comprehensive resources that can facilitate the development of at least the most common NLP tasks. The development of corpora requires a great amount of effort. In the case of diachronic corpora, a more specialized degree of expertise is necessary, but that is scarce. We will continue collaboration in activities that develop such resources, as is the case of RoDia. Such tasks constantly meet new challenges and require technical support to aid with the progress of the development.

At the same time, more Romanian linguistic resources must be created, especially in the area of semantics. This is a segment in which Romanian is deficient. These resources may enable the development of a semantic-based dependency parser and also create new research opportunities regarding Romanian in other areas of NLP, such as Machine Translation. We will continue our on-going collaborations in activities that target these types of developments. We will continue to develop Ro-PAAS, and we will sustain our implication in the SSPR project. The semantic annotation is not an easy task; an alignment with current research in the field, to use an international applied system of annotation and not to neglect the Romanian approach in this domain, the Romanian FrameNet (Trandabăț, 2010), is highly desired.

Another interesting research activity is merging the developed NP chunker with the VP chunker, in what would initially be a rule-based clause splitter. A clause is basically composed of a verbal group and noun phrases. Going further with such a development, a unified GGS grammar could also display the constituent structure of clauses, and even expose aspects of the discourse structure of sentences. We envision that semantic information is required for achieving high quality results for Romanian, especially in the challenges imposed by prepositional attachment.

This thesis focuses on describing our involvement in the development of tools for processing Romanian language, which have fructified, besides various conclusions, into an important achievement: GGS, a flexible tool that fits a wide range of use cases.

We have contributed to the development of Romanian linguistic tools and resources. We have discovered new methods of constructing diachronic resources by means of semi-automatic

processes. We have developed GGS, which can be used for developing other tools, for querying corpora and for analysing linguistic data. We have ventured into GGS Grammar Inference, and found that it promises to enable exposing the deep structure of surface information, which is represented as sequences of tokens. We have enabled access to NLP enthusiasts and researches to a basic set of Romanian linguistic tools: POS tagger, Dependency parser and NP chunker.

Pure information is a continuous flow of unstructured data. Light, sound, atomic particles of the universe, they are all waves of information, entering the human mind through our senses. For some of the processes that have sustained life and enhanced its quality, life forms have evolved to discern information into structured units, a process that is commonly referred to as *generalization*. This is the basic process that allows information to be interpreted, processed further, stored and transmitted. This process is not necessarily a trait of the mind but rather a characteristic of life itself: we see information being transmitted through various means such as DNA, hormones, pheromones, electrical impulses, visual and auditory streams etc. in all kinds of life forms, at every scale of the spectrum, i.e. from the microscopic cellular life to the macroscopic interaction in social species, among which, human beings are by far the most evolved.

Communication is fundamental to life. It is required by the internal components of an organism and also between organisms. Besides the usual internal biological communication required to sustain life, an emotional system has evolved as a solution to an elevated life quality, especially in mammals. Additionally, in humans, a complex verbal language has evolved. The verbal language is not used only for communication between individuals, but also for intrapersonal communication, as a tool for developing and refining highly organized thoughts, which one can then express externally or hold onto internally as core beliefs. We believe that verbal language is the fundamental reason for the increased capability of the human mind to process and structure information into complex networks of abstract generalization units. This has allowed the development of analytical thinking and collaboration of individuals in complex tasks to develop tools for enhancing life quality and to explore the nature of our world.

Not all forms of communication require the information to be first generalized and interpreted before transmission. The interpretation can be performed entirely by the receiver, as is the case of a human being watching a photograph, which is a snapshot of raw visual data. But in the case of linguistic communication, information is first generalised into words and structured into sentences before being delivered. There is much debate about what a word is. We like to believe that words are signals that have been conventionally and naturally assigned by large groups of individuals to common internal information structures of the brain. These signals are not necessarily in the form of sound.

In oral communication, language is aided by tonality, rhythm, tempo and volume variation. In face-to-face communication it is also aided by visual cues, which compose a consistent segment of what is referred to as non-verbal communication – a complex, separate language in itself. Human kind has invented the form of transmitting language through writing. Textual communication generalizes any non-digitized information such as tonality or visual cues into a consciously agreed language expressed through what is known as *punctuation*. Written communication uses only abstract symbols, which have been mapped to phonemes by convention. Identical concepts are mapped by different languages to different acoustic signals and combining rules. Words may also be mapped by individuals that are deaf using the so-called *sign language*, a form of communication that makes use of hands and finger gestures and facial expressions. Blind-deaf individuals make use of tactile signs as a form of communication. Blind individuals can read tactile writings.

Language can be transmitted in various ways, through various channels. There is research conducted even in the direction of brain-to-brain communication. No matter the channel, its structure is bound by the same principles of generalization, which are fundamental to life. Linguistic information is segmented into units that are labels of complex, common concepts. Units may be tied by relations, and multiple units of information may be encapsulated together as one. Networks of hierarchical nodes of information are thus formed.

Besides language, which may be seen as an evolutionary invention, the computer is probably the next most important achievement of the human kind. It is a tool that is changing social life in a fundamental way. It is an extension of the mind, in a sense. Computers cannot yet process complex unstructured data, but can process, store, transfer and quickly access huge volumes of structured information, traits that the human mind lacks. Combined with language, information technology has enabled instant communication of linguistic messages over huge distances and to make knowledge accessible from anywhere on earth. This has transformed the previous model of our world, i.e. isolated societies communicating through very narrow conduits, into a global nexus of communication.

NLP is the field of artificial intelligence that aims to bridge the gap between the evolutionary invention of verbal language and the recent advances in IT. NLP promises to enable real time automatic translation between languages, contributing further to the global nexus of communication. It promises to enable a natural language based interaction with computer software. It promises to enable computers to take over some of our most boring tasks, allowing and also forcing human kind to focus on creative thinking while conquering new perspectives over the universe. But what fascinates me the most is that it promises to empower computers with capabilities closer to those of the human mind, which would allow them to discern huge amounts of information to discover root causes and solutions to the world's biggest problems.

GGS is just a little step in this grand vision. It directly addresses the blueprint of language by enabling traversing information which is structured by the principles of generalization. This is the reason why it can be used in so many NLP tasks. GGS patterns are currently promoted as a form of manually controlling parsing behaviour. GGS Grammar Inference represents the path towards learning such patterns, with the hope that further research and developments will lead to a new realm in NLP.