UNIVERSITÀ DEGLI STUDI DI SALERNO

Dottorato in Scienze della Comunicazione

Exploring Formal Models of Linguistic Data Structuring

Enhanced Solutions for Knowledge Management Systems Based on NLP Applications

Federica Marano

Supervisor
Prof. Annibale Elia

Coordinator
Prof. Alessandro Laudanna

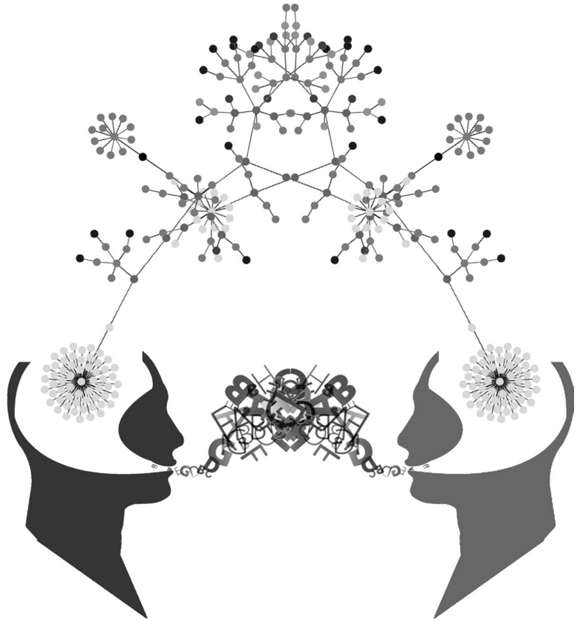
X Ciclo – Nuova Serie
2008-2011

UNIVERSITÀ DEGLI STUDI DI SALERNO

Dottorato in Scienze della Comunicazione

Exploring Formal Models of Linguistic Data Structuring

Enhanced Solutions for Knowledge Management Systems Based on NLP Applications



Federica Marano

Supervisor
Prof. Annibale Elia

Coordinator
Prof. Alessandro Laudanna

X Ciclo – Nuova Serie
2008-2011

Index of acronyms

Computer Assisted (Aided) Translation	CAT
Content Management System	CMS
Data Mining	DM
Decision Support System	DSS
Finite State Automata	FSA
Finite State Transducers	FST
Frame Element	FE
Generative Grammar	GG
Transformational-Generative Grammar	TGG
Generative Transformational Linguistics	GTL
Information and Communication Technology	ICT
Information Extraction	IE
Information Retrieval	IR
Knowledge Management	KM
Knowledge Management System	KMS
Latent Semantic Analysis	LSA
Latent Semantic Indexing	LSI
Lexicon-Grammar	LG
Linguistic Resource	LR
Machine Translation	MT
Multi Word Units	MWU
Natural Language Processing	NLP
Part Of Speech	POS
Principal Component Analysis	PCA
Semantic Web	SW
Singular Value Decomposition	SVD
Slow Intelligent System	SIS
Text Classification	TC
Text Mining	TM

Contents

Abstract

The principal aim of this research is describing to which extent formal models for linguistic data structuring are crucial in Natural Language Processing (NLP) applications. In this sense, we will pay particular attention to those Knowledge Management Systems (KMS) which are designed for the Internet, and also to the enhanced solutions they may require. In order to appropriately deal with this topics, we will describe how to achieve computational linguistics applications helpful to humans in establishing and maintaining an advantageous relationship with technologies, especially with those technologies which are based on or produce man-machine interactions in natural language.

We will explore the positive relationship which may exist between well-structured Linguistic Resources (LR) and KMS, in order to state that if the information architecture of a KMS is based on the formalization of linguistic data, then the system works better and is more consistent.

As for the topics we want to deal with, frist of all it is indispensable to state that in order to structure efficient and effective Information Retrieval (IR) tools, understanding and formalizing natural language combinatory mechanisms seems to be the first operation to achieve, also because any piece of information produced by humans on the Internet is necessarily a linguistic act. Therefore, in this research work we will also discuss the NLP structuring of a linguistic formalization Hybrid Model, which we hope will prove to be a useful tool to support, improve and refine KMSs.

More specifically, in section 1 we will describe how to structure language resources implementable inside KMSs, to what extent they can improve the performance of these systems and how the problem of linguistic data structuring is dealt with by natural language formalization methods.

In section 2 we will proceed with a brief review of computational linguistics, paying particular attention to specific software packages such Intex, Unitex, NooJ, and Cataloga, which are developed according to Lexicon-Grammar (LG) method, a linguistic theory established during the 60's by Maurice Gross.

In section 3 we will describe some specific works useful to monitor the state of the art in Linguistic Data Structuring Models, Enhanced Solutions for KMSs, and NLP Applications for KMSs.

In section 4 we will cope with problems related to natural language formalization methods, describing mainly Transformational-Generative Grammar (TGG) and LG, plus other methods based on statistical approaches and ontologies.

In section 5 we will propose a Hybrid Model usable in NLP applications in order to create effective enhanced solutions for KMSs. Specific features and elements of our hybrid model will be shown through some results on experimental research work. The case study we will present is a very complex NLP problem yet little explored in recent years, i.e. Multi Word Units (MWUs) treatment.

In section 6 we will close our research evaluating its results and presenting possible future work perspectives.

Keywords

Knowledge Management System, Natural Language Processing, Linguistic Formal Model, Hybrid Formal Model.

Foreword

The core of this research project is to achieve computational linguistics applications helpful to humans in establishing and maintaining an advantageous relationship with technologies, especially with those technologies which are based on or produce man-machine interactions in natural language.

The ideal exploitation milieu of these applications is the Internet, which in the digital and new media era, particularly in the "www" era, is more and more becoming a crucial cognitive tool used for describing, ranking and linking data, researching, building analytic and communicating environments, and for many other different purposes. Internet is today the most traditional repository for documents, images, multimedia and other reusable resources, covering an extremely vast range of topics, and theoretically offering online answers to everyone having an IP connection.

But the Internet is an immense world, composed by about one milliard pages. Actually, only a very low percentage of them is retrievable by users, and this limitation leads to observe that statistically it is impossible to correctly retrieve the information needed at a first attempt. This impasse is mainly due to the fact that the search engines predisposed to treat the content of all Internet pages chunk and index information without "understanding" them: all the words which compose information are considered as mere sequences of characters delimited by blanks, while those word combinatory rules on which meaning production is founded are almost always disregarded.

In this sense, two most basic and even somewhat trivial considerations are to make; firstly, no information could be stored on the Internet without natural language; secondly, by definition information are originally created only for human consumption and aren't machine readable. This leads us to pose specific questions, as for instance: how can we correctly retrieve semantic information from the Internet, and mainly using which tools? How much important is natural language studying and processing in the definition and building of such tools? Is there any real possibility of bringing search engines to "understand" the information they are supposed to chunk and index?

If we consider that our research is just at its starting point, then we must admit that these questions cannot still have precise answers. It is only possible to state that in order to structure efficient and effective IR[1] tools, understanding and formalizing natural language combinatory mechanisms seems to be the first operation to achieve, also because any piece of information produced by humans on the Internet is necessarily a linguistic act.

Actually, many technologies promise a new paradigm of semantic information sharing. The grandest vision of this is the Semantic Web (SW), in which the enormous body of data available on the Web will be organized in a way that allows it to be indexed by its meaning, not just by its form (Allemang, 2006).

[1] Information Retrieval is a set of studies aiming at developing techniques and methodologies to correctly retrieve electronic information, and only the needed information. But, if we consider the Web, "information" is a too generic term; we intend metadata about documents, about structured storages, relational databases, etc. For an accurate and complete definition of "Information Retrieval" see Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, *Introduction to Information Retrieval*, Cambridge University Press, 2008 (also online on http://nlp.stanford.edu/IR-book/).

Tim Berners-Lee, pathfinder of the SW, proposed it as a solution (Berners-Lee, 2001), as a sort of machine-readable Web that uses intelligent agents in order to guide the user to specific desired information and to help him carrying out some operations automatically. Therefore, in a recent online interview published by the Italian newspaper "La Repubblica", Berners-Lee says that he his «…glad to design, among other things, tools for the SW that are based on the concept of linked data. SW is based on data, while search engines work with hypertext documents. The challenge for search engines has been to try to create a structure where there was no structure, to instill order and meaning where there was no order or meaning, while with data order and meaning are already present. When you have data in an archive, they are already well ordered and well structured and have a much more defined content. As for the Web, finally more people are now realizing the value of "linked open data". This will allow the new Web to be smarter» [2]. Only in this sense the concept of SW can be considered as a cognitive tool, i.e. only if it facilitates access to information on the Web.

Moreover, and still considered utopian by many, SW is not just a line of research that strives to make information available and accessible to all. Rather, it attempts to do so in a way that is effective and understandable to all. And obviously, the best way to retrieve information using semantic criteria is exploiting the power of natural language in its full characteristics.

Furthermore, the definition of Information Society, which includes the concept of SW, has always been configured as a set of theories and

[2] Excerpt from Luna R. *Così ho regalato il web al mondo*. "La Repubblica" 14 novembre 2011, http://www.repubblica.it/tecnologia/2011/11/14/news/intervista_berners_lee-24969134/?ref=HRERO-1).

best practices necessary to make information a competitive advantage in today's Knowledge Societies[3]. If Information Society always preached the overthrow of Digital Divide, i.e. of that gap existing between those who have means and resources to access to the Internet and those who for different reasons (technological, economic, infrastructural) have not, then SW could be considered as the overthrow of Cognitive Divide, i.e. the gap existing between Web ordinary and experienced users and mainly consisting in the denial of a set of best practices useful to make relevant information accessible to all in few clicks and without too much effort. But the focus has been put on Digital Divide overthrowing, and consequently most of efforts done have been essentially technocentric. At any rate – as said Clotilde Fonseca, Minister of Science and Technology of Costa Rica during 2010 – «today's digital divide is strongly linked to the cognitive divide. It is related to the way in which people are able to understand, learn, express, produce, share, collaborate, create, and innovate using technology. This demands the activation of intellectual and knowledge acquisition skills and competencies with growing levels of diversity and complexity» (Fonseca, 2010).

[3] «The aim of the information society is to gain competitive advantage internationally through using Information Technologies in a creative and productive way. An information society is a society in which the creation, distribution, diffusion, use, integration and manipulation of information is a significant economic, political, and cultural activity. The knowledge economy is its economic counterpart whereby wealth is created through the economic exploitation of understanding. People that have the means to partake in this form of society are sometimes called digital citizens.» For a generic point of view see http://en.wikipedia.org/wiki/Information_society; for a more specific treatment see http://ec.europa.eu/information_society/index_en.htm.

Introduction

One of the main features of contemporary society is certainly the vast and varied amount of information being produced today. If we dwell only on a few examples of written information, and omit those which are audio-visual in a strict sense, we can take as an example the huge number of books published each month in the world; also, the elevate quantity of printed material, newspapers and deepening magazines; the great number of Web sites that are opened every day in the Internet; or all the information network users exchange through emails and social network short massages, such as Tweets and Facebook posts.

This large amount of information that overwhelms us every day goes by the name of Information Overload. The spread of Information Overload definition is due to Alvin Toffler (1970), an American futurist writer. Although the concept was born long before the diffusion of the Web, actually it is very appropriate to define the overload of information inputs we receive from the Internet, i.e. from the technological tool with which we interact more.

Information Overload is the most negative aspect of the concepts of "democratization of the Web" and "accessibility to information resources". Among others, the causes which originate it are not only the ever-increasing production of information, but above all the natural multiplication of this information through the large quantity of Internet-based channels, media and technologies. So, if on one hand the Internet and all the new technologies allow anyone having an IP connection to dispose of readily-available billion information, on the other all this create a sort of “vacuum effect”, because having too much information is equal to haven’t any. In algorithmic terms, we can say that too much data to examine, or too many possibilities to explore, stretch excessively the processing of a given problem, moving away from the correct solution, or even making more complex its individuation. In fact, an algorithm programmed to analyze a problem and to choose only the best solution – or at least a satisfactory set of heuristic solutions – is forced to examine all possible answers, which thereby increases exponentially the resolution time of the problem. This would be well known to George Bernard Dantzig, the mathematician who introduced the simplex algorithm (simplex method) inside linear programming (Dantzig, 1940; Dantzig *et al.*, 1951). Breaking down all possible solutions which must be taken into account, simplex algorithm leads to the correct solution of a problem only if it takes places a particular optimal condition which is contemplated by the problem itself. Unfortunately, this type of algorithm does not work with all kinds of problems, and in some cases the solutions to consider may grow exponentially[4]. After Dantzig, whose masterly studies are still valid and extensively used, many other theories have been developed to try improving problems resolution[5]. One of the latest trends is the one concerning Slow Intelligent Systems (SISs), in which during the decision-making process the system itself is not constrained by the “time” factor. Unlike classical algorithms, a

[4] For a more detailed biography on George Dantzig see http://en.wikipedia.org/wiki/George_Dantzig; for more information on his works, linear programming and simplex algorithm see: http://www.stanford.edu/group/SOL/dantzig.html; and also Albers, Donald J. and Constance Reid. *An Interview of George B. Dantzig: The Father of Linear Programming*. College Mathematics Journal. Volume 17, Number 4; 1986 (pp. 293-314).

[5] See http://www.cs.pitt.edu/~chang/1635/c11SIS/si01.htm

SIS does not stop at the first solution found, but ciclically experiences new ones, adapting itself to environments, spreading knowledge and exchanging information flows with other systems, in order to improve its performance. It is a kind of Zen approach, implemented in an automatic troubleshooting. A SIS is a system that (i) solves problems by trying different solutions, (ii) is context-aware to adapt to different situations and to propagate knowledge, and (iii) may not perform well in the short run but continuously learns to improve its performance over time (Chang, 2010; Colace *et al*., 2010).

But an even superficial analysis could demonstrate that neither simplex method nor SISs are suitable to solve the problem of Information Overload. As previously stated, Dantzig's simplex method works on the presence of particular optimal conditions contemplated by the problems itself. If we need to have efficient and effective IR, i.e. natural language interpretation performed by machines, then we must not forget that natural language expressions are essentially lexicon-based and syntax-governed combinatory strings, and that the interpretation problems they pose are only solvable thanks to optimal conditions they do not directly contemplate – but that can anyway be formalized, as we will see in the following chapters. At the same time, due to the complex role played in the interpretation of natural language expressions by non-compositionality and pragmatics, it is possible to state that SIS aptitude to adapt to new environments would be heavely questioned by the fact that each natural language string could be a unique and not replicable environment in itself. This would strongly affect SIS possibility to ciclically experience new solutions[6].

[6] In this sense, we could also call into question the concept of creativity introduced by Chomsky (1968). This might be useful to emphasize that with regard to both simplex

The problems just highlighted become even more urgent under the pressure of Information Overload, which actually amplifies and reproduces them exponentially, making them almost tending to infinity. In this regard, an observation seems necessary: in order to overrun Information Overload, IR must be supported by accurate and effective solutions as far as natural language formalization methods are concerned. Moreover, these formalization methods must be focused not only on the structural aspect of language strings, but also and necessarily on the relationship between form (structure) and content (meaning) of the same strings. This means that the creation of semantic-based IR systems (or better, KMSs) can be possible only after the finding of a really effective method of formalization of natural language, capable of analyzing the relationship between form and content of strings.

This makes even more evident the necessity to solve the problems related to the management of information flow. If Information Overload is one of the most negative factors that affect human choices within problem solving and decision making activities, then there is the need to develop efficient KMSs and Decision Support Systems (DSSs) to help the human beings in filtering information, in order to receive only the relevant solutions needed and to provide more tools to pick the right decision (optimal or at least satisfactory) for a specific problem.

method and SISs, digital infinity, or the idea that natural language strings are the result of the infinite use of finite means, would exclude *a priori* the possibility of having strings (i.e. environments) likely to be grouped into homogenous sets. This seems to be true as regards the final part of speech production, i.e. the strings physically composed of words. On the contrary, as we will see, it is possible to group natural language strings on structural basis, studying and classifying their characteristics according to the syntactic and predicate-based concepts of transformational structures introduced by Harris (1952) and co-occurrence and selection restriction introduced by Gross (1981).

The overload phenomena just mentioned with reference to problem solving, decision making, and KM can be widely debated in different fields. However, in this work we wish to emphasize features and peculiarities which are essentially related to the world of new technologies on the Web. Actually, if we focus precisely on this question and restrict the fields of application, then we discover that the most controversial points are without doubt connected to IR. We can say, without the risk of venturing wrong assumptions, that the problem of IR represents the origin and root cause of all other problems related to Knowledge Management (KM). To take a concrete proof, let us consider a user who must make a decision on an ordinary aspect such as the online purchase of a TV-set, and let us imagine that he is trying to understand what information better suit his needs. He will have to make choices about the size (in inches or centimeters), the type of screen (Plasma, LCD, LED, 3D), the color(s) of the frame, and other specifications. At first glance, this example can be a problem easily to solve, but actually it is so only in two cases, namely: 1. the user is an expert on TV-sets; 2. even not being an expert on TV-sets, the user knows in advance what he is looking for. In point of fact, these two conditions do not occur very easily, and the most common scenario is the one in which a user find himself at the mercy of too much information to make a decision. Consequently, he is forced to go and visit a retailer to personally ask information on the best TV-set to buy.

For this and other ordinary e-commerce problems, there are many technologies that seek to eliminate barriers to the online purchase of products. Putting aside the purely economic and legal aspects, if we focus on the simple choice of a product, we find that there are modern user profiling systems, or automated systems, which “interrogate” users to understand their needs and provide tailored products, customized for them.

But what lies behind all these profilers and the personalization of this trend? What makes possible the progress of these and other technologies related to human-machine interaction? The basic problem, once again falling onto IR, essentially concerns information architecture, or better: how data are structured, on the base of which criteria they are ranked, if these classifications meet the fundamental principles of completeness and consistency, and so on. Therefore, from all this comes the consideration that well-structured files or databases by themselves already produce meaningful and relevant information. We have previously mentioned Berners-Lee's assertion that *when you have data in an archive they are already well ordered and structured and they have a more distinct meaning*. Then, everything brings us back to the organization of information, to the indistinct and continuous stream of data which must be structured to be usable.

Let us get deeper into the problem of information architecture and make some preliminary considerations. It is an ancient man's need to digitize, make discrete, classifiable and ordered the indistinct flow of knowledge which by its nature is analog. Leaving aside the numerous possible digressions on this subject and focusing only on the Web, it is not difficult to define it as a network of databases and repositories containing a huge quantity of information which are not strighforwardly accessible to the public, currently somewhat disconnected, and the extraction of which is often undermined by the presence of (more or less technological) proprietary query interfaces. When Berners-Lee speaks of files of structured data which are already sufficient in themselves to meet the information needs of a user, he clearly refers to another important concept, i.e. the one of "linked data". Therefore, in his opinion, the first operation to achieve would be making data (whatever form of data: text, tables, images and so on) accessible by anyone. Actually, open data are one of the main goals to achieve according to the best practices dictated by the W3C Consortium[7] as for SW. The subsequent step to make the Web a truly surfable hypertext is to link all the data it contains. Only by respecting these rules it will be possible to get a first Web based on linked data, efficient and usable, in which documents, images, and data are all recognizable, open, well classified, and easily manageable. But when we talk of billions of data, it is not so easy to abide by these criteria. Especially because one of the biggest problems which may be experienced is the difficulty in finding and adopting unique and standard parameters. Therefore, standards utilization becomes an essential practice, and from its inception until now, the W3C Consortium has been using all his energy to convert Internet users to these recommendations. W3C standards define an Open Web Platform for application development that has the unprecedented potential to enable developers to build rich interactive experiences, powered by vast data stores that are available on any device. The boundaries of the platform continue to evolve, but its full strength relies on several technologies which W3C and its partners are creating, including CSS, SVG, WOFF, the SW stack, XML, and a variety of APIs. W3C develops these technical specifications and guidelines through a process designed to maximize consensus about the content of a technical report, to ensure high technical and editorial quality, and to earn endorsement by W3C and the broader community. The application field concern: 1. *Web Design and Applications*: it involves the standards for building and rendering Web pages, including HTML5, CSS, SVG, Ajax, and other technologies for

[7] The World Wide Web Consortium (W3C) is an international community that develops open standards to ensure the long-term growth of the Web. http://www.w3.org/.

Web Applications (“WebApps”). This section also includes information on how to make pages accessible to people with disabilities (WCAG), internationalized, and work on mobile devices; 2. *Web Architecture*: it focuses on the foundation technologies and principles which sustain the Web, including URIs and HTTP; 3. *Semantic Web*: in addition to the classic “Web of documents” W3C is helping to build a technology stack to support a “Web of data”, the sort of data you find in databases. The term “Semantic Web” refers to W3C’s vision of the Web of linked data. SW technologies enable people to create data stores on the Web, build vocabularies, and write rules for handling data. Linked data are empowered by technologies such as RDF, SPARQL, OWL, and SKOS; 4. *XML Technologies* including XML, XQuery, XML Schema, XSLT, XSL-FO, Efficient XML Interchange (EXI), and other related standards; 5. *Web of Services*: it refers to message-based design frequently found on the Web and in enterprise software. The Web of Services is based on technologies such as HTTP, XML, SOAP, WSDL, SPARQL, and others; 6. *Web of Devices*: W3C is focusing on technologies to enable Web access anywhere, anytime, using any device; 7. *Browsers and Authoring Tools*: Web agents are intended to serve users when designing browsers and authoring tools, as well as search engine bots, aggregators, and inference engines.

The list of all main application fields covered by W3C recommendations on Standards helps us to understand that the work of organizing information is long and arduous; many factors must be taken into account to achieve a complete and consistent data classification, and every community of Web developers and researchers must do its part to make improvements. As already mentioned, this work will highlight all the potential improvements which can be achieved to efficiently organize data from a particular and specific point of view: the one concerning natural language.

In the following sections we will cope in detail with all the topics only hinted at so far. In section 1 we will see how to structure language resources implementable inside KMSs, to what extent they can improve the performance of these systems and how the problem of linguistic data structuring is dealt with by natural language formalization methods. In section 2 we will proceed with a brief review of computational linguistics, paying particular attention to certain software packages such Intex, Unitex, NooJ, and Cataloga, developed according to LG method, a linguistic theory established during the 60's by Maurice Gross. In section 3 we will describe some of the works written to monitor the state of the art in Models of Linguistic Data Structuring, Enhanced Solutions for KMSs, and NLP Applications for KMSs. In section 4 we will cope with problems related to natural language formalization methods, describing mainly TGG and LG, plus other methods based on statistical approaches and ontologies. In section 5 we will propose a Hybrid Model to use in applications of NLP in order to create effective enhanced solutions for KMSs. Specific features and elements of hybrid model will be shown through some results on experimental research work. We focused on a language problem that is very complex and yet so little explored in recent years, MWUs treatment. In section 6 we will close this dissertation evaluating our results and presenting possible future work perspectives.

The Relationship between Linguistic Resources and Knowledge Management Systems

1 *Well-structured Linguistic Resources for effective Knowledge Management Systems*

Before going into detail and explore the positive relationship which may be established between well-structured LRs and KMSs, we will make a brief introduction on KM. Subsequently, we will examine the reasons why the performance of a KMS can be improved if based on the embedding of logically-formalizied data and language resources.

The term KM was originally born with meaning connotations related to the corporate world, primarily as a set of management skills-oriented transfer of knowledge within companies, that is to say that transfer of skills and competencies which arises from the experience of a company and its employees, and which makes such company competitive on the market. It is the explicitness of business know-how when it is transformed into well-summarized procedures that must be transmitted and spread in a company and in its components to ensure that it remains active and competitive.

In more general terms, KM comprises a range of strategies and practices used in an organization to identify, create, represent, distribute, and enable adoption of insights and experiences. Such insights and experiences comprise knowledge, either embodied in individuals or embedded in organizations as processes or practices. But from our point of view, if we focus our analysis only on Web activities, we notice the presence of very interesting developments. More recently, especially in the second half of the 20th century and due to the always increasing use of computers, KM has began concerning specific adaptations of technologies such as knowledge bases, expert systems, knowledge repositories, group DSSs, as well as intranets. KM is one of the hottest topics today in both industry and information research world. In our daily life, we deal with huge amount of data and information. Data and information do not become knowledge until we do not succeed in extracting its value out of them. This is the very reason why we need KM.

The History of KM starts during the 70's. A number of management theorists have contributed to its evolution: Peter Drucker introduced the idea that the concepts of "information" and "knowledge" could be considered as organizational resources (Drucker, 1969), while Peter Senge started talking of "learning organization" (Senge *et al.*, 1994). On the contrary, during the 80's, it became more evident that knowledge could represent a competitive asset to explicit within professional competences. Subsequently, it was developed the concept of managing knowledge, which relied on the work done in artificial intelligence and expert systems. Finally, the International Knowledge Management Network (IKMN) went online in 1994, but the most important growth in KM was introduced in the popular press, by Ikujiro Nonaka and Hirotaka Takeuchi, who wrote The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation (1995), which became the most widely read work to date on KM subject.

Today there are many technologies highly correlated to KM, and they cover all the steps of its life cycle. Knowledge is acquired or captured using tecnologies as intranets, extranets, groupware, Web conferencing and document management systems. Successively, an organizational memory is formed by refining, organizing, and storing knowledge using structured repositories such as for instance data warehouses. Then knowledge is distributed through different education tools, training programs, automated knowledge based systems, expert networks. Knowledge is applied or leveraged for further learning and innovation via organizational memory mining and expert systems application, such as DSSs. Each segment of these stages is enhanced by effective workflow and project management.

The future of KM consists of ad-hoc software that will develop knowledge-aware enterprise management systems. Knowledge collaboration portals will be created in a way to efficiently transfer knowledge in an interdisciplinary and cross functional environment. Information systems will evolve into artificial intelligence systems that use intelligent agents to customize and filter relevant information. New methods and tools will be developed for KM driven e-intelligence and innovation. Therefore, we can imagine that multiple corporate databases will merge into large, integrated, multidimensional knowledge bases designed to support KMSs in terms of competitive intelligence and organizational memory. These centralized knowledge repositories will optimize information collection, organization, and retrieval. They will offer knowledge enriching features that support the seamless interoperability and flow of information and knowledge. These features may include: the incorporation of video and audio clips, links to external authoritative sources, content qualifiers in the form of source or reference metadata, and annotation capabilities to capture tacit knowledge. Content will be in the form of small reusable learning objects and associated metadata that provides contextual information to assist KM reasoning and delivery systems.

This brief introduction on KM helps us in deepening our initial hypothesis, i.e. the fact that if the information architecture of a KMS is based on the formalization of linguistic data, then the system works better and is more consistent. There are at least two reasons to justify this hypothesis. The first is explained by the fact that a KMS developed starting from linguistic resources is based and can rely upon concrete and tangible formalizable data, as lexicon, morphology, syntax, and formal semantics. The second is explained by means of all those logic properties and principles which are specific to natural language, as for instance semantic roles[8] and logical linguistic operators[9].

An attentive examination of all the elements just mentioned may help us in giving substance to the connection existing between language resources and KMS. When we affirm that a KMS is more effective if developed on the basis of linguistic data we state that such data, being present and tangible, can minimize the margins of potential errors in-

[8] In linguistics, semantic roles are used to describe meanings attached to complements on the basis of the process expressed by predicates inside sentences, or to adopt an Harrisian terminology, by operators with reference to the arguments they select (Harris 1976). The notion of "semantic role" has been developed as part of linguistic theories attempting to interconnect the syntactic and semantic components of language. It is linked to the notion of "syntactic function" and "case", but it cannot be confused with them: while functions and cases are defined by syntax, semantic roles are in principle independent. In the context of generative linguistics, semantic roles are within the deep structure of a language, that is to say within the organization of concepts and relations, while the functions and structure of cases are within the surface structure, that is to say within the representation of this organization in the grammatical forms of a particular language. For a lexicon-grammar based configuration of semantic roles, see (Gross 1981).

[9] Boolean operators are the most renowned and used logical linguistic operators. They take their name from George Boole, an English mathematician of the first half of the 19th century which formalised the binary logic that underlies modern computers. As far as search tools are concerned, the main and most commonly used boolean operators are AND, OR, NOT, and NEAR.

herent in specific KM tasks, that is to say in IR and more specifically, in applications as Question Answering routines in which an automatic query system (a search engine) is used. Greatly simplifying the structure of the whole system, which will certainly be more complex, we can say that a local search engine (which does not scours the entire Web but only a portion ot it, or a piece of knowledge which is pre-established and circumscribed) should be formed by a database which contains all the information and the possible relations between them, and a software system, a crawler, which allows the scanning of the database using a set of preliminary queries. It would be natural to think of queries formulated by means of keywords following as much as possible the criterion of plausibility to the concept/information sought after. However, we must not forget that users "think" information they seek in a way almost ever different from the one in which developers have "thought" the same information. This exemplifies one of the main limits of a relational database, that is to say the dissonance existing between the criterion of classification adopted and the information needs to express in natural language. In similar situations, elements such as lexicon, morphology, syntax and formal semantics could come to the help. For example, a lexical ontology connected to the database would allow us to find the concepts of *clean energy* or *renewable energy*, whereas the input keyword was focused on the concept of *bioethanol*. This would possible because, within an ontology, the concepts of *clean energy* and *renewable energy* would be synonymous, and both hypernyms of the term *bioethanol*. This example shows how the formal semantics emerging from the logical-semantic relations among concepts can become a distinctive feature and a strong disambiguation tool. Another innovative element could be implemented starting from the concept of linked data. If all information repositories relating to energy were connected for instance not only to the database of our previous example, but also to other databases, such as online encyclopedias on energy, FAQ repertoires on the same topic, and so on, then the crawler could simultaneously examine multiple lexical resources and return relevant information.

It is also possible to envisage the setting up of more complex queries, in which keywords correspond to compound words and not to simple words or free word groups. There is a clear difference between these types of expressions. As a typical differentiation made in computational linguistics, as well as a typical setup of NLP software, from a formal point of view we may observe that:

- Simple words, as for instance *panel*, *card*, *chair*, are sequences of characters delimited by blanks, or by a blank and a diacritic symbol.
- On the contrary compound words, as for instance *solar panel*, *credit card*, *rocking chair*, are composed by two or more simple words separated by blanks or diacritic symbols. Also, as complete sequences, compound words are delimited by blanks, or by a blank and a diacritic symbol.
- Free word groups, as for instance *huge panel, torn card, white chair*, have the same formal aspect as compound words.

But the main defference existing among this three formal elements – and which can be essentially deduced from their formal differences – concern their use within concrete acts of signification: while simple words and free word groups must be necessarily contestualized to acquire a precise meaning, compound words almost always have a predefined frozen or fixed one. This automatically means that when used as query keywords, compound words are by definition more effective in retrieving information than simple words and/or free word groups.

There are many problems related to the accuracy which must be used in this type of distinction (Downing, 1977; Silberztein, 1993; Sag et al., 2001; Girju, 2005; Laporte et al., 2008; De Bueriis G. and Elia A. eds., 2008); this is the main reason why one of the most relevant problems with IR software systems is the correct processing of compound words, or better MWUs, also known as complex lexical units [10]. The shortcomings are mainly due to the fact that such units are often considered as extemporaneous combinations of words retrievable by means of statistical routines. On the contrary, several linguistic studies, also dating back to the ‘60s, show that MWUs, and mainly compound nouns, as already stated are almost always fixed meaning units, with specific formal, morphological, grammatical and semantic characteristics. Furthermore, these units can be processed as dictionary entries, thus becoming concrete lingware tools useful to achieve efficient semantic IR. Another important problem is due to the fact that up to today there is still no universally agreed definition or term for the concept of MWU. In literature we often find concurrent terms such as “multiword”, “multiword expression”, “fixed expression”, “idiom”, “compound word”, and “collocation” used by many authors of different theoretical schools or following distinct NLP approaches, but all these terms, even though ambiguous in themselves, all refer to the same concept of “string of words in which all elements are related one to the other”. For instance, collocations are defined as expressions consisting of two or more words that correspond to some conventional way of saying things (Manning and Schütze, 1999), that have the characteristics of syntactic and semantic units, with exact and unambiguous meanings or connotations

[10] To properly investigate this topic, in section 5 we will propose some experimental research work on MWUs treatment.

which cannot be derived from the meanings or connotations of its components (Choueka, 1998). Also Sinclair (1991) considers collocations as typical expressions of a linguistic combination principle not bounded by grammaticality constraints.

In earlier LG framework[11], the most essential features of what we call MWUs were non-compositionality and semantic opaqueness. Maurice Gross (1986) uses the term compound word to refer to a string composed of several words the meaning of which cannot be computed from its elements. Recently, the significance of compositionality has changed, and the term MWU has evolved in such a way that it can also be referred to non-idiomatic units, being now used to refer to various types of linguistic entities, including idioms, compounds, phrasal verbs, light or support verb constructions, lexical bundles, etc. LG scholars have long been studying MWUs, and the practical analytical formalization has been done for several languages. Besides, in (D'Agostino & Elia, 1998) MWUs are considered as part of a continuum in which combinations can vary from a high degree of variability of co-occurrence of words (combinations with free distribution), to the absence of variability of co-occurrence. They identify four different types of combinations of phrases or sentences, namely: 1. with a high degree of word co-occurrence variability, i.e. with free internal distribution, compositional and denotative meaning; 2. with a limited degree of word co-occurrence variability, i.e. combinations with restricted internal distribution; 3. with no or almost no word co-occurrence variability, i.e.

[11] A brief review on LG will be presented in the section 2.2, for more specifications see also http://en.wikipedia.org/wiki/Operator_Grammar; http://en.wikipedia.org/wiki/Zellig_Harris; http://fr.wikipedia.org/wiki/Lexique-grammaire; http://infolingu.univ-mlv.fr/ (click on "Bibliographie"); http://it.wikipedia.org/wiki/Lessico-grammatica.

combinations with fixed internal distribution; 4. without any word co-occurrence variability.

Relations between these mentioned classes can be interpreted not only as relations between distinct classes, but also as relations between poles of the continuum. We give here some examples of these combination classes: (for combinations at point 1.) verbal structures: (*Max*, *Ugo*, *your nephew*,...) *looks at* (*a book*, *the river*, *Eva*,...); nominal structures: (*clean*, *dirty*,…) *water*; adverbial structures: *with* (*elegance*, *love*, *devotion*,...) (for combinations at point 2.) verbal structures: (*Max*, *Ugo*, *your nephew*,...) *dries* (*the clothes*, *the laundry*,…); nominal structures: (*mineral*, *sparkling*, *natural*,…) *water*; adverbial structures: *from one* (*moment*, *day*, *year*,...) *to the other*; (for combination at point 3.): verbal structures: (*Max*, *Ugo*, *your nephew*,...) *bends his elbow*; nominal structures: *heavy water*, *arsenic water*; adverbial structures: *in no uncertain terms*; (for combination at point 4.) proverbs: *walls have ears*.

From a semantic point of view, and with reference to communication processes, we observe that types (c) and (d) may also have “idiomatic” interpretations, or rather interpretations that are not semantically compositional (i.e. not coming from a compositional computation of the meanings of each lexical element). Probably, some of these fixed and idiomatic combinations are the result of metaphoric and metonymic drifts which have been lexicalized. Starting from these assumptions, we may deduce that the use of the four mentioned combination types originates from the need for incisive and immediate communication processes rather than for ordinary ones. While metaphor and metonymy, as any figure of speech, involve an additional operation of decoding and interpretation, fixed and idiomatic combinations are used as a single block: they are semantic shortcuts, and it is not necessary to know the meaning of each element of the linguistic sequences they are conveyed by. But it is important to stress that in LG, all these types of lexical entries can be formalized, coherently inserted inside linguistic databases (i.e. electronic dictionaries), and used within NLP routines, such as for instance IR and parsing. Each type of MWU may need to follow a different formalization method. There is the morphological aspect of MWUs (i.e., the morphology of composition) that weights considerable for morphologically-rich languages and remains a highly challenging task. From a lexicographical point of view, MWUs with a specific grammatical function and an autonomous meaning need to be registered in dictionaries in a systematic way (Laporte & Voyatzi, 2008), i.e. as autonomous lemmata and not, as often is the case in traditional dictionaries, as examples of use of head nouns or adjectives. As far as electronic-dictionary lemmatization is concerned, a clear distinction between MWUs with a high degree of variability of co-occurrence among words and those with a limited or no variability of co-occurrence among words (compound words, idiomatic expressions, and proverbs) should be made. This is one of the most critical issues in the description of natural languages. For example, there is a relevant difference in Italian between *colletto bianco* and *colletto celeste* (which only has the meaning of *blu collar*). The first has to be lemmatized since it has also the specific meaning of *white collar worker*, and has distinctive formal, morphogrammatical and lexical properties, i.e.: a) it is invariable, as it does not accept any insertion or addition, for instance **colletto molto bianco* (**very white collar worker*); b) is a singular masculine compound noun only referring to a "human being", with *colletti bianchi* as its masculine plural form. On the contrary, *colletto celeste* does not possess these characteristics, being a free nominal group, therefore not necessarily lemmatizable. This is quite a simple example of the difference between opposite poles in the continuum. Sometimes, however, MWUs are much more difficult to classify and describe. For example, the Italian MWU *editto bulgaro* (*Bulgarian edict*) and *elezione bulgara* (*Bulgarian elections*) are on the edge between the status of compound words and that of free nominal groups. This is a problem that occurs most frequently with compound words. According to Elia *et al*. (2008), an accurate identification of compound words must be based on the following criteria:

- Semantic atomicity. If the exact meaning of a nominal group cannot be deduced from the meaning of its components, the nominal group must be lemmatized (=> it is therefore treated as a compound noun. This happens with the already mentioned *colletto bianco*, but also with *teste di cuoio* (members of a special anti-terrorist police team), *casa chiusa* (*brothel*) *Guerra Fredda* (the proper noun *Cold War*), in which each element of the compound participate in the construction of a complete and non-literal meaning.
- Distributional restriction. If certain constituents of the nominal group, which by the way, belong to certain natural distributional classes, cannot be freely replaced, then this distributional restriction must be acknowledged by classifying the series of nominal groups in a lexicon, which again, amounts to treating it as a compound noun. For instance, the above-mentioned examples of *colletto bianco* and *colletto celeste* follow this criterion.
- Institutionalization of the usage. Certain nominal groups, even those that are semantically and distributionally "free", are used in a quasi-obligatory manner, to the detriment of other potential syntactic constructions that are just as valid, but are never used. The Italian expression *in tempo reale* (a loan translation of the English *in real time*) is an example for this criterion, which use

> in Italian seems to be unmotivated if we take into consideration that the antonym **in tempo irreale* (**in unreal time*) is not used at all. These criteria allow identifying a larger group of compound words than it is normally and traditionally assumed for a language.

Indeed, Computational Linguistics developed lots of measures of association; an association is any relationship between two measured quantities that renders them statistically dependent. These measures are useful to quantify the strength of the bond between two or more words in a text. But many methods which rely on frequentist or probabilistic approaches to retrieve MWUs do not take into account strings of words referred to as "single meaning units" in a proper way, even if highly frequent, thus resulting in loss of information. On the contrary, our approach aims at building a linguistically motivated identification of MWUs, on the basis of a systematic and exhaustive formalization of natural language.

2 *A Brief Review of Computational Linguistics*

Computational Linguistics is first of all the fusion of these two words: “Linguistics” and “Computational” (see Figure 1). So it primarily is the study of human languages through the use of computers. According to one of the first definitions of computational linguistics, it is the study of computer systems for understanding and generating natural language (Grishman, 1986.).

The history of Computational Linguistics is closely connected to the development of the digital computer; it was born as an hybridation between Linguistics and Computer Science, but it is very important to state it as an interdisciplinary field. Its theoretical foundations cover also Artificial Intelligence (a branch of *computer science* aiming at computational models of human cognition), Cognitive Science, Logics, Psycholinguistics, Mathematics, Philosophy, Engineering among others. Computational linguists are interested in providing computational models of various kinds of linguistic phenomena. These models may be “knowledge-based” (“hand-crafted”) or “datadriven” (“statistical” or “empirical”).

The commitment to “simulation of behaviour”, shared by Artificial Intelligence and a relevant part of Computational Linguistics, makes them also share the effort for “cognitive modelling” of different human behaviours, including the use of language. This is probably one of the

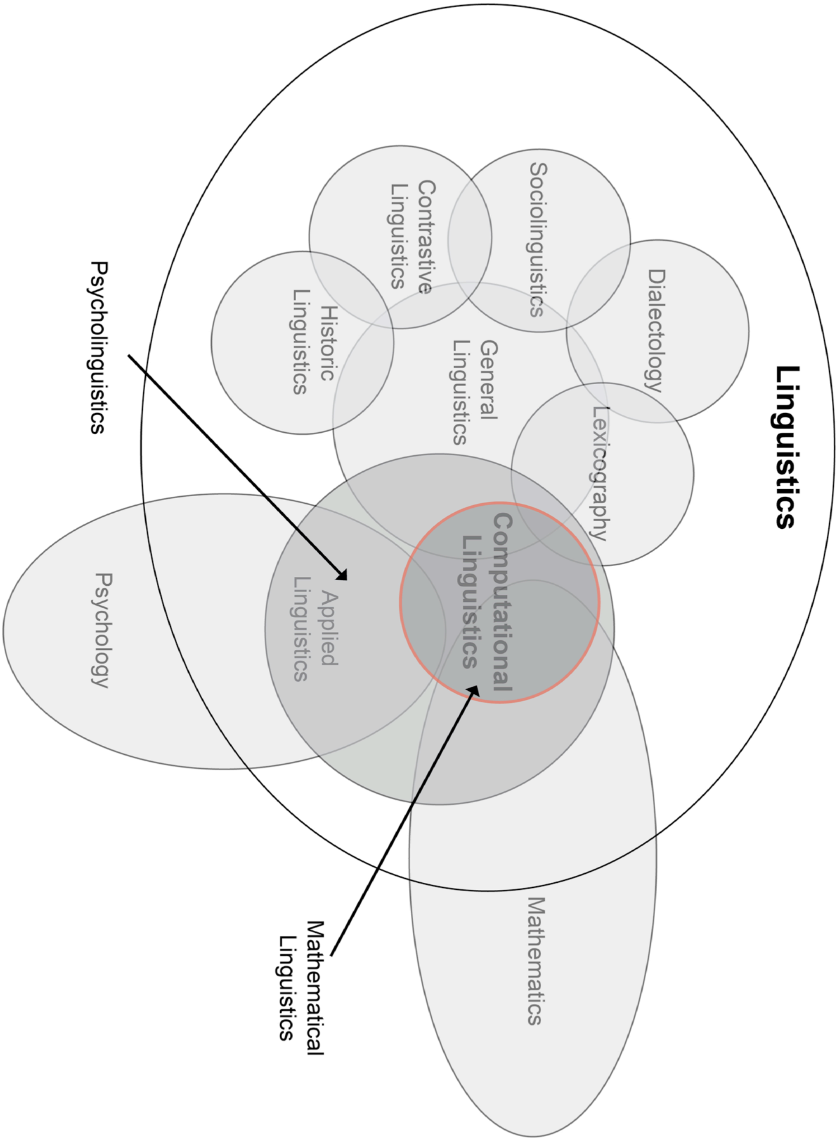


FIGURE 1. Structure of linguistic science.

reasons why Linguistics appears in the set of sciences originally interested in the arising of the new discipline called Cognitive Science[12].

The close link between so diverse disciplines, and coming from such different fields of science, stems from the fact that since its inception computer science has been concerned with natural language which, moreover, is that faculty that best characterizes the human beings and their nature. If we focus on the activities in which the computer are used to achieve important research aims, we find studies on simulation, robotics, human/computer interfaces, and many other fields that especially today are continuously evolving. Linguistics, for its part, has seen almost immediately in computing the set of powerful tools and techniques of calculation that would help to manage complex cognitive theoretical systems on language, enormous amounts of data (large-scale corpora), and that would finally allow automatic analysis of texts within a reasonable time.

This is why computational linguistics is primarily the study of language but with the support of computer science tools and techniques. It is the analysis of language with the help of the computers, allowing data processing in large amounts and in a short time. At the same time, it cannot be simply regarded as a discipline that deals with language processing, because it is not limited to design artificial systems capable of intelligent performance; in such cases, computers are just a tool.

It is important to consider that computational linguistics also makes use of techniques which do strictly relate to Computer Science, and which apply and implement manual or classical means of investigation: this is the case of style, statistics and corpus linguistics.

[12] For more specifications see http://www.cognitivesciencesociety.org.

On such basis, the following definition of Computational Linguistics seems very actual and accurate:

> «Lo studio del linguaggio con l'ausilio del calcolatore. Anche se di fatto le ricerche di linguistica computazionale sono spesso intrecciate con quelle di intelligenza artificiale, si usa distinguere tra linguistica computazionale ed elaborazione (automatica) del linguaggio naturale (ELN) perché la prima non persegue anzitutto la realizzazione di sistemi artificiali capaci di prestazioni intelligenti in rapporto al linguaggio, ma invece la conoscenza del linguaggio stesso, e usa spesso il calcolatore come strumento di verifica di teorie linguistiche indipendenti. Inoltre, fanno parte della linguistica computazionale (ma non dell'ELN) ricerche che usano tecniche informatiche "non intelligenti", come quelle di stilistica computazionale e in generale quelle basate sull'elaborazione (anche con strumenti statistici) di corpora lessicali, in vista della realizzazione di vocabolari, concordanze, ecc. La linguistica computazionale è peraltro impegnata in tutti i settori della ricerca linguistica teorica, dalla sintassi alla pragmatica e all'analisi del discorso, attraverso la costruzione di sistemi che realizzino teorie o frammenti di teorie linguistiche.» [13].

[13] Beccaria G.L. (ed.), 1994. "The study of language by means of computers. Actually, even if computational linguistics research are often intertwined with those of artificial intelligence, a distinction is usually made between computational linguistics and (automatic) natural language production (NLE) because the former does not pursue as its primary goal the realization of artificial systems capable of intelligent performance in relation to language, but rather the knowledge of the language itself, and it often uses computers as a tool for testing independent language theories. In addition, part of computational linguistics research (but not of NLE) use "dumb" computer techniques, as those of computational stylistics and generally those based on lexical corpora development (including the use of statistics), in view of the creation of dictionaries, concordances, etc.. Computational linguistics is also engaged in all areas of theoretical linguistic research, from syntax to pragmatics and discourse analysis, through the construction of systems which implement theories or fragments of linguistic theories." (Translation by the editor).

Therefore, computational linguistics could also be considered as a general and theoretical linguistics, because it treats morphology, syntax, semantics, pragmatics, but with a plus: it uses computer and computer-formal-statistical techniques.

Computational linguistics and computer science share some of the same fundamentals. The idea of parsing, for example, is a central characteristic in the design of any programming language compiler, as well as being the "building block" of NLP. So the first real application of Computational Linguistics was in the area of Machine Translation (MT) (at that time better known as mechanical translation). Expectations of an intelligent machine arise almost immediately in the United States in the '50s, during the Cold War, to use computers to automatically translate texts from foreign languages, particularly Russian scientific documents, into English.

The illusion was to believe that a transfer grammar [14] and a bilingual vocabulary would be sufficient to achieve good MT.

Early attempts to design an intelligent machine able to perform in a totally automatic translation from one language to another had an approach based on building a pivot language.

The pivot language was a sort of abstract and semantically unambiguous language that would provide 1 to 1 correspondence of word and concept.

The procedure included a text in the target language 1 (the language), which was first translated into pivot language and then it was generated a text in the target language 2 (the target language).

[14] In order to achieve an efficient automatic translation process, (Harris 1954) creates transfer grammars which formalize differences among languages in terms of maximum similarities (or minimum differences). From a strict formal point of view, this method is one of the most profitable ever structured, even if we will see that it was doomed to failure due natural language specific idiosyncrasies.

Soon, these first attempts clashed with the problems of ambiguity. Blatant errors in MTs made with the first applications, as it amusingly happened with the proverb *The spirit is willing, but the flesh is weak* (дух бодр, плоть же немощна, an allusion to Mark 14:38) was translated into Russian and then back to English, resulting in "*The vodka is good, but the meat is rotten*" (спирт, конечно, готов, но мясо протухло).

Between '50s and '60s the idea spread that Computational Linguistics was effortless to cope with. Indeed, many researchers predicted that MT problems could be solved in about three years. Although they used different approaches, mostly hand-coded rules/linguistics-oriented ones, the three-year project continued for ten years, yet producing no good result, despite the significant amount of expenditure. By the '60s, computational linguistics was placed under the larger realm of computer research of artificial intelligence. After the initial hype in the early '70s, a dark era came in MT, due the fact that many started believing that it was impossible. Consequently, research in computational linguistics were almost abandoned. In any case, since '70s, when language technology reached a state of maturity such as to allow the realization of some applications, Engineering has been interested in some of the language processing techniques, and it soon appeared that the approach introduced by engineers was certainly less theoretically and cognitively interesting, but more effective in many ways. By now, we can say that while Computational Linguists were, and are, more interested in the correctness and plausibility of their models, Engineers were, and are, more interested in the usability of tools and techniques, even at the cost of some "dirty" solutions (Ferrari, 2004).

Between '70s and '80s there was a slow revival of Computational Linguistics. Some research activities revived, but the emphasis was still on linguistically oriented tools and solutions which coped with small toy problems with weak empirical evaluation, at least until '90s when the computing power increased substantially. Statistics takes over other approaches, data-driven statistical approaches with simple representation win over complex hand-coded linguistic rules; Jelinek (Brown *et al*. 1988) says: "Whenever I fire a linguist our MT performance improves".

Nowadays, as it will be shown throughout this research work, statistics alone is not enough to handle the numerous tasks computational linguistics usually copes with, also considering that formal linguistic models even quite dissimilar one from the other have become part of the real core of this so complex discipline.

Over the years and with the proliferation of paradigms and computational models of language, a curious and paradoxical phenomenon has been created. Currently, it is computational linguistics which "helps out" general linguistics in the development and progress of the analysis. This happens for instance with the development of large lexical corpora, which is a very useful activity in the study of isolated phenomena of syntax, morphology and lexicography.

Computational linguistics can be divided into major areas depending upon the medium of the language being processed, whether spoken or textual; and upon the task being performed, whether analyzing language (recognition) or synthesizing language (generation).

Speech recognition and speech synthesis deal with how spoken language which can be understood or created using computers. Parsing and generation are sub-divisions of computational linguistics dealing respectively with taking language apart and putting it together. MT remains the sub-division of computational linguistics dealing with having computers translate between languages.

Some of the areas of research that are studied by computational linguistics include [15]:

- Computational complexity of natural language, largely modeled on automata theory, with the application of context-sensitive grammar and linearly-bounded Turing machines.
- Computational semantics comprises defining suitable logics for linguistic meaning representation, automatically constructing them and reasoning with them.
- Computer-aided corpus linguistics.
- Design of parsers or chunkers for natural languages [16].
- Design of taggers like POS-taggers (part-of-speech taggers).
- MT as one of the earliest and most difficult applications of computational linguistics draws on many subfields.
- Simulation and study of language evolution in historical linguistics/glottochronology.

2.1 *A Short Survey on Some Main Computational Linguistics Subfields*

In the following pages, we will specifically examine only few major subfields of computational linguistics as Corpus linguistics, Parser designs, Tagger designs, MT and Logic Designs.

[15] This classification is taken from http://en.wikipedia.org/wiki/Computational_linguistics.

[16] We here assume that in this case is not taken into consideration the great methodological and analytical difference that exists between a (syntactic) parser and a chunker. For instance, in one of the NLP software environments that we will describe in 2.3, chunking (i.e. tokenization) and parsing are two distinct part of a modular procedure in which also indexing is provided for.

«Corpus linguists analyze how everyone...exersize[s] their minds in language» (Lancashire, 2000). In essence, this subfield of computational linguistics combines several analysis techniques, like text analysis and cybertext theory, to look at representative samples of language to determine patterns. Using statistics, researchers can observe authors' habits and generate hypothetical texts.

As for Natural Language Parser Designs, parsers attempt to break up natural language sentences into their smallest understandable "chunks" – words, punctuations, and special symbols (Nanduri & Rugaber, 1995). Subsequently, through the help of a dictionary and a set of grammar rules, parsers determine the structure of input sentences and attempt to determine their meaning. Although on the surface this may appear a simple task to achieve, it becomes more complex due to certain language-specific intricacies, such as different uses of the same word in English, i.e., tear, broke, or feet. Generative and Transformational Grammar (Chomsky, 1957; 1965) was the original linguistic theory that deeply influenced this kind of studies, but its computational interpretation gives rise to a number of different models, with different both technical and theoretical impacts. The key problem to solve is to reach a logical (deep) structure of the sentence, such as to satisfy the constraint of being mapped onto some sort of semantic (executable) representation. Anyway, Chomsky's transformational grammar does not offer a direct solution to this problem, leaving free space to several interpretations.

Another important subfield regards Tagger Designs for Natural Languages, in which the majority of work deals with the design of parts-of-speech taggers (Abney, 1997). While employing the background knowledge from natural language parsers, parts-of-speech tagging, also known as "parts-of-speech disambiguation", attempts to uncover the meaning of words based on statistical (stochastic) or rule-based algorithms. So far, however, parts-of-speech taggers are stuck in the development phase, particularly in situations in which a tagger encounters an unknown word (Van Guilder, 1995).

In the previous pages, we have already defined MT as the initial subfield of computer linguistics and as the use of computers to translate from one natural language to another, without any human intervention. This translation process can be divided into two categories, since there are «some approaches that require manual knowledge entry, while others make use of automatic training procedures» (Knight & Marcu, 2005). In either process, however, the words of the text, as well as the punctuation and symbols, are tokenized – or segmented, in MT-research terms – and then translated into the desired language.

Finally, an interesting new area concerns Logic Designs for NLP. According to Alshawi (Alshawi, 1994), this relatively new subfield of computational linguistics deals mainly with speech recognition. This logic system uses a «grammar and lexicon to produce a set of logical forms». The grammar for the logic design, like any grammar, is expressed as a set of syntax and semantic rules. Human utterances are recorded and then translated into first-order logic expressions, which are then passed onto a translation device, and finally tested against a theorem prover.

The work of many researchers in computational linguistics allows people to interact and communicate with machines using natural language.

As said, the main aims are recognition, interpretation, translation and language generation, while remaining complex and controversial discipline, a sort of discipline "container" in which you can easily pour theories, models and other techniques taken from other fields, it should be noted that there are lots of areas in which some achievements have been reached and that those progress conduce to develop, for instance, essential application of computational linguistics as texts analyses (written texts and spoken texts), aided translation, speech recognition (automatic dictation software systems), texts generation (written texts and spoken texts), and many others applications in specific sectors of Web, Telephony and Communication, Help Desk, E-Health and E-Government, Tutoring System, Disability Support (deaf, visually impaired), Conversational Agents.

2.2 *Lexicon-Grammar, a Frame for Computational Linguistics*

In this section will deal with LG, a formal analysis framework principally based on structuralism and which is largely complementary with many pragmatic developments of computational linguistics, in particular those concerning NLP. This complex of theories, methods and tools was born during the ‘60s from the research made on natural language by Maurice Gross (Gross, 1968; 1989), a French linguist who had initially trained as an engineer.

Actually LG is one of the most profitable and consistent methods for natural language formal description. It was originally set up for French and subsequently developed for and applied to Italian by Annibale Elia (Elia, 1984), Emilio D’Agostino and Maurizio Martinelli (EMDA, 1981). In the course of time, it also has been widely applied to several different languages. Nowadays, it describes both Indo-European languages (French, Italian, Portuguese, Spanish; English, German, Norwegian; Polish, Czech, Russian, Bulgarian; Greek) and others (Arabic; Korean; Malagasy; Chinese; Thai). Its descriptive methodology has also reached important results in the domain of automatic textual analysis and parsing, with the creation of software and lingware fully oriented towards NLP, such as INTEX and UNITEX[17], and more recently NooJ[18] and Cataloga. We also can recall the exixtence of several LG studies on specialty languages (see also Gross, 1975; Elia, 1984).

LG analytic methodology is based on specific mathematical models of language (Harris, 1982; Schützenberger in Gross et al., 1973), and its main goal is to describe syntax by formalizing all mechanisms of word combinations closely related to concrete lexical units and sentence creation. This description is built not on statistic-based rules and algorithms, but on the analysis of words co-occurrence, distribution and selection restriction observed inside simple sentences[19] by means of predicates syntactic-semantic properties. This analytical method is mainly based on Zellig Harris' concepts of Operator-Argument Grammar (Harris, 1982)[20], and of transformational rules (Harris 1964). In such sense, starting from the bloomfieldian notion of morpheme and from the method of commutation or equivalence between different morphemes (Bloomfield, 1933), LG transformational rules can high-

[17] More information on the website http://www-igm.univ-mlv.fr/~unitex/.

[18] See http://www.nooj4nlp.net/pages/nooj.html.

[19] In LG, simple sentences are defined as the minimal linguistic meaning contexts in which co-occurrence, selection restriction and distribution can be analyzed. More specifically, a simple sentence is a context formed by a unique predicative element (a verb, but also a name or an adjective) and all the necessary arguments selected by the same predicate in order to obtain an acceptable and grammatical sentence. For more specification on simple sentence definition see Gross (1968).

[20] As for this topic see also the Valency Theory developed by the French linguist Lucien Tesnière (1953; 1959).

light mutual relationships between simple sentences (active/passive, positive/interrogative, etc.) having different formal aspects but similar meanings. Unlikely well-known formalist and syntax based linguistic theories such as Chomsky's deep grammar and its various offsprings (Chomsky, 1957; 1965)[21], LG approach assumes that the formal description of natural language has to start from the observation of lexicon and of lexical entry combinatory behaviours, encompassing both syntax and lexicon. Thanks to this approach, LG allows the international community of linguists to get a complete, empirical and exhaustive description of natural languages by means of a large data set consisting of tables of syntactico-semantic properties of thousands of lexical entries (mainly Verbs, Nouns, Adjectives).

For these reasons, LG can be considered an empirical methodology founded on the observation and recording of linguistic data. More specifically, it is an empirical approach in the sense that it is not based on any a priori reasoning. Also, it may be viewed as a manually-based methodology functional to the development of tailor-made linguistic resources. In this sense, LG can be useful in NLP applications structuring, especially in Web-knowledge procedures focused on IR goals and KMS creation. As a matter of fact, the linguistic resources[22] developed according to the LG framework rely on the empirical observation of data, which are located and isolated in concrete contexts. Successively, data are classified on the basis of common characteristics and behav-

[21] However, in the Minimalist Program Chomsky acknowledges that the phrase structure is also derived from the lexicon, thus there is a projection of the lexicon upon the syntax (Chomsky, 1993; 1995).

[22] LG main linguistic resources include electronic dictionaries and local grammars.

iours (i.e. word distributions, co-occurrences, predicate-based selection restrictions, syntactic government, allowed transformations). Therefore, it is very likely that the classes which LG detects may be composed by a single element with unique features[23].

As previously stated, LG range of analysis invests lexicon, and especially the concept of Multiword Units as "meaning unit", "lexical unit" and of "word group", for which LG identifies four different combinatorial behaviours (see De Bueriis et al., 2008):

- Combinations with a high degree of variability of co-occurrence between words. In this case we have combinations based on open distribution with a compositional and signified meaning;
- Combinations with a low degree of variability of co-occurrence between words. In this case we have combinations based on constrained distribution;

[23] Formerly, Joseph Harold Greenberg, one of the most original and influential linguists of the twentieth century, introduced empirical methodologies in linguistic researches. Greenberg, a pioneer in the development of linguistics as an empirical science, founded his work directly on quantitative data taken both from a single language or from a wide range of languages. His chief legacy to contemporary linguistics is in the development of an approach to the study of language – typology and univerals – and to historical linguistics. Yet he also made major contributions to sociolinguistics, psycholinguistics, phonetics and phonology, morphology, and especially African language studies. According with him, following an empirical and functionalist method means to found researches on a sample of languages as wide as possible. On the contrary, a logical-deductive and rationalist method, such as the Chomsky's one, founds researches on the properties of a single tongue. Greeenberg deals almost immediately with Linguistic Universals, and specifically in reference to them he highlights the difference between the two types of approaches, the empiricist and the rationalist, preferring the first one. Therefore LG with its survey methodology based on the analysis and the formal classification of data, is very close to the positions of Greenberg. For a complete profile on Geenberg see the obituary of William Croft (University of Manchester) that appeared in *Language*, vol. 77 no. 4 (2001), pp. 815-830.

- Combinations with zero or almost zero degree of variability of co-occurrence between words. In this case we have combinations based on fixed distribution;
- Combinations without variability of co-occurrence between words. In this case we have proverbs.

Relations between these mentioned classes could be interpreted not only as relations between separated classes, but also as relations between poles of a continuum. We give here some examples of these combination classes:

a) (combinations at point 1.)

- Verbal structures: (*Max*, *Mary*, *your nephew*,...) *looks at* (*a book*, *the river*, *Eva*,...)
- Nominal structures: (*clean*, *dirty*,…) *water*
- Adverbial structures: *with* (*elegance*, *love*, *devotion*,...)

b) (combinations at point 2.)

- Verbal structures: (*Max*, *Mary*, *your nephew*,...) *dries* (*the clothes*, *the laundry*,…)
- Nominal structures: (*mineral*, *sparkling*, *natural*,…) *water*
- Adverbial structures: *from one* (*moment*, *day*, *year*,...) *to the other*

c) (combination at point 3.)

- Verbal structures: (*Max*, *Mary*, *your nephew*,...) *bends his elbow*
- Nominal structures: *heavy water*, *arsenic water*
- Adverbial structures: *in no uncertain terms*

d) (combination at point 4.)

- Proverbs: *Walls have ears*

From a semantic point of view and for disambiguation tasks, we observe that types (c) and (d) may also have “idiomatic” interpretations, or rather interpretations that are not semantically compositional (i.e. not coming from a compositional computation of each lexical element meaning). Probably, some of these fixed and idiomatic combinations are the result of metaphoric and metonymic drifts, which have been lexicalized.

Starting from these assumptions, we may deduce that the use of the four mentioned combination types originates from the need for incisive and immediate communication processes rather than for ordinary ones. While metaphor and metonymy, as any figure of speech, involve an additional operation of decoding and interpretation, fixed and idiomatic combinations are used as a single block: they are semantic shortcuts, and it is not necessary to know the meaning of each element of the linguistic sequences they are conveyed by. It is important to stress that in LG, all types of lexical entries can be formalized, coherently inserted inside linguistic databases (i.e. electronic dictionaries), and used within NLP routines, as for instance IR and parsing, LRs, built in this way and managed using the above-mentioned criteria, are useful to effective semantic tagging.

2.3 *Lexicon-Grammar: Resources, Tools and Software for Computational Linguistics*

Resources and tools used by LG consist of:

1. matrix tables describing predicates syntactic-semantic properties;
2. electronic dictionaries morphologically and semantically tagged;
3. local grammars in form of Finite State Automata and Finite State Transducers (FSAs/FSTs).

Using rows and columns, LG matrix tables describe syntactic properties of predicates – i.e. not only of verbs, but also of nouns, adjectives. Each row corresponds to a predicate, and each column represents a formal property. Rows may describe both distributional and transformational properties, using the sign "+" or "-" the presence of which means that the predicate, respectively, can or cannot accept a specific property. We give here an example of an Italian lexicon-grammar table, in which the first three left column show properties for N_0 (the logical subject), while the nine left columns refer to different constructions that the verb can or cannot take:

N0 =: Num	N0 =: il fatto Ch F	N0 =: V1 Comp	V	N0 V che F a N2	F0 V che Fcong a N2	N0 V di V0 Comp a N2	N0 V di V2 Comp a N2	il fatto Ch F a N2	N1 =: se F o se F	N1 =: N1 V1-inf Comp	N0 V a N2 di N3	Passivo
-	+	+	Apportare	+	-	-	+	+	-	-	-	+
+	-	-	Articolare	+	+	+	+	-	+	-	-	+
+	-	-	Assegnare	-	+	-	+	+	-	-	-	+
+	-	-	Asserire	+	-	+	-	+	-	-	-	+
+	+	+	Assicurare	+	-	+	-	+	-	-	-	+
+	-	-	Attribuire	+	-	-	+	+	-	-	-	+

TABLE 1: Example of LG table

On the contrary, before describing LG electronic dictionaries, it is necessary to make a terminological and formal disctinction between electronic and computerized dictionaries. Actually, it has been highlighted (Vietri et al. 2004) that the term “computerization” has somehow confused the two categories. Print modernization processes require that the texts of conventional paper dictionaries are typographically composed on computerized media, but this computerization process does not affect the content of these dictionaries, which remains unchanged. So, both paper and computerized dictionaries are only used by humans having a solid and already existing expertise.

A fairly recent example of computerized dictionaris are those CD-Rom versions on which the original content of paper dictionaris is enhanced by multimedia solutions (word pronunciation, hypertext navigation) that greatly implement both presentation and consultation. However, the lemmata description in these dictionaries is not formally homogeneous as the one of LG electronic dictionaries, and in the following pages we will see how this lack of fromalism prevents KMSs from using computerized dictionaries as dependable linguistic resources. On such basis, it is also possible to state that while LG electronic dictionaries may be successfully used as linguistic engines embedded in NLP routines, as for instance automatic textual analysis, this is not possible with computerized dictionaries, which are not fully (re)usable for NLP purposes.

In fact, LG electronic dictionaries are built according to strict formal rules, are only used by computers within specific software routines, and are managed by specialized human users. Data included in such dictionaries are formalized by means of codes which are not intelligible to common readers.

LG electronic dictionaries are all part of DELA[24] system, a lexical database[25] homogeneously structured and in which the morphogrammatical characteristics of lexical entries (gender, number and inflection) are formalized by means of distinctive and not-ambiguous alphanumeric tags.

[24] Acronym from Dictionnaire Électronique of LADL (Laboratoire d'Automatique Documentaire et Linguistique).

[25] The term *database* is intended according to the most common meaning of Informatics, from both the theoretical and the practical point of view.

These dictionaries have different formal and content characteristics, which produce different formal classifications. Consequently, DELA electronic dictionaries may be of two types:

- simple word dictionary (DELAS 120,000 ca. canonical words and DELAF 1,200,000 inflected words), which includes lexical units as *home* and *chair*, i.e. semantically autonomous and formed by sequences of characters delimited by blanks or by a blank and a punctuation mark;
- terminological compound word dictionary (DELAC 154,000 ca. canonical compound words and DELACF 460,000 ca. inflected compound words subdivided in dictionaries of specific knowledge domains), which includes lexical meaning units as *nursing home*, and *rocking chair*, i.e. lexical units composed by of two or more simple words and characterised by a global meaning which may also be non-compositional [26].

Compound word electronic dictionaries mostly lemmatize terminological entries [27]. The development of terminological electronic dictionaries is achieved by a manual data entry procedure which is supervised by linguists and domain experts.

[26] For more on this topic, see paragraph 2.2.

[27] Unlike simple words, which are often polysemic and ambiguous, compounds have a polysemic rank almost always near to zero, which is an important characteristic as far as terminological and specialized languages are concerned. Besides, as already stated, from a formal and morphological point of view there are concrete differences between simple and compound words, which must be necessarily accounted for both with reference to NLP routines and when building linguistic databases.

As a sample, we give here an extract of the Italian Electronic Dictionary of Medicine. Its exploitation in this work is fully explained in Section 5:

quarto ventricolo, N + Genere = m + Numero = s + Class = AN + Term = MED

pronto soccorso, N + Genere = m + Numero = s + Class = AN + Term = MED

malattie infettive, malattia infettiva, N + Genere = f + Numero = p + Class = NA + Term = MED

agenti patogeni, agente patogeno, N + Genere = m + Numero = p + Class = NA + Term = MED

flora residente, N + Genere = f + Numero = s + Class = NA + Term = MED

Additionally, DELAC-DELACF may also be multilingual, so becoming useful for other specific NLP applications, such as MT systems. We give below sample strings extracted from the Italian-English compound word dictionary of Medicine:

ubriachezze patologiche, ubriachezza patologica, N + Genere = f + Numero = p + Class = NA + Term = MED + Eng = pathologic intoxication, pathologic intoxication, Number = s+ Class = AN

uditi cromatici, udito cromatico, N + Genere = m + Numero = p + Class = NA+ Term= MED + Eng = chromatic audition, chromatic audition, Number = s+ Class = AN

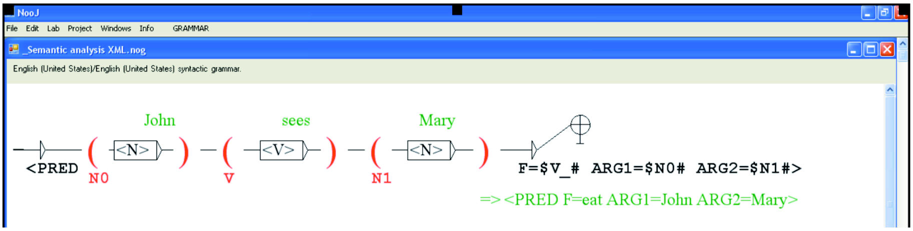
uditi residui, udito residuo, N + Genere = m + Numero = p+ Class = NA + Term = MED + Eng = residual hearing, residual hearing, Number = s + Class = AN

It is worth noting that the compound word domain dictionaries collected in the DELAC system represent 180 different domain fields, each one of which is identified by means of a specific semantic tag. Among these dictionaries, the most important are those of Informatics and Computer Science (approx. 54,000 entries), Medicine (approx. 46,000 entries), Law (approx. 21,000 entries) and Engineering (approx. 19,000 entries). Subset tags are also previewed for those domain sectors which include specific subsectors. This is the case with Engineering, for which a generic tag ING is used, while nine more explicit tags are used for Acoustic Engineering (ING ACUS), Aeronautics and Aerospace Engineering (ING AER), Chemical Engineering (ING CHIM), Civil Engineering (ING CIV), Mechanical Engineering (ING MECC), Mining Engineering (ING MIN), Naval Engineering (ING NAV), Nuclear Engineering (ING NUCL) and Oil Engineering (ING PETROL). A same formalization was used for Physics, which has been given a generic tag FIS plus more specific tags for Atomic Physics (FIS ATOM), Nuclear Physics (FIS NUCL), Physics of Plasma (FIS PLASMA), Solid-State Physics (FIS SOL) and Particle Physics (FIS PART).

Local grammars are the third type of resources used by LG in NLP routines, or more generally in natural-language based IR applications, as for instance automatic responders, question answering, and so on. These grammars are called local because they account only for particular grammatical features of a given language; they are used to parse texts on the basis of the syntactic descriptions they cover, which essentially encompass transformational rules and distributional behaviours (Harris, 1957). Local grammars are constructed in form of FSA/FST[28],

[28] An FST has an input part in which are included the text sequences to process, and an output part in which processing results are given. On the contrary, an FSA can be defined as a special case of FST that doesn't produce any result (i.e. it has no output) (Silberztein, 1993; 2002). FSAs are typically used to locate morph-syntactic patterns in corpora; they also can extract matching sequences in order to construct indices, concordances, etc.

i.e. either deterministic or non-deterministic oriented graphs in which specific formalisms are used to first recognize and subsequently disambiguate, tag and rewrite sets of text sequences. FSTs/FSAs are useful to automatically recognize and parse any kind of text. Figure 2 gives an example of a graph describing the specific syntactic behaviour of the verb *to see* and having as output a text tagged in XML:


NooJ
File Edit Lab Project Windows Info GRAMMAR
_Semantic analysis XML.nog
English (United States)/English (United States) syntactic grammar.
John
sees
Mary
<N>
<V>
<N>
N0
V
N1

```
<PRED F=$V_# ARG1=$N0# ARG2=$N1#>
=> <PRED F=eat ARG1=John ARG2=Mary>
```

FIGURE 2. This grammar recognizes any sentence in which there is a structure with an Operator (V) plus two Arguments.

Within parsing procedures, similar grammars are applied during the input phase, while the output consists of a text annotated with tags reusable in subsequent IR routines.

To construct and test electronic dictionaries and local grammars, LG uses two software packages: NooJ[29] and Cataloga.

NooJ is an NLP environment built by Max Silberztein, which embeds large-coverage dictionaries and grammars, and which parses even sizeable corpora in real time. It also includes tools to create and maintain large-coverage lexical resources, as well as morphological and syntactic grammars. Dictionaries and grammars are applied to texts in order to

[29] See http://www.NOOJ4nlp.net/pages/NOOJ.html.

locate morphological, lexical and syntactic patterns and tag simple and compound words. NooJ can build complex concordances, with respect to all types of Finite State and Context-Free patterns. Therefore NooJ users can easily develop extractors to identify semantic units in large texts, such as names of persons, locations, dates, technical expressions of finance, etc. NooJ can process texts and corpora made of hundreds of text files. Lexical, syntactic and semantic annotations can be inserted in the text in cascade, without destroying the text. NooJ linguistic engine is multilingual and it can perform Harris's transformations in cascade, giving NooJ the power of a Turing Machine. The morphological and the syntactic engines are integrated: this makes it possible to perform morphological operations on words while performing a syntactic transformation. As of today, NooJ can process a dozen languages, including some Roman, Germanic, Slavic, Semitic and Asian languages, as well as Hungarian. Its dictionaries and grammars are extremely simple objects to build, requiring the learning of no complicated formalism. It is a complex NLP environment in which it is possible to automatically read digitized texts, retrieving from them specific linguistic patterns in the form of concordances. NooJ's engine is based on the DELA system of electronic dictionaries, on LG syntactic tables and on FSAs/FSTs, by means of which it parses texts.

Cataloga is a software built by Annibale Elia, Alberto Postiglione and Mario Monteleone (Elia, Postiglione & Monteleone, 2010). It uses LG terminological DELAC-DELACF dictionaries and it is based on the matching between these dictionaries and digitized texts. It is actually configured as a stand-alone software which can be integrated in Web sites and portals to be used online. The main linguistic goal of this software is to extract terminology from a given scientific or technological text and to automatically determine – without human reading – the main knowledge domains it deals with. From a computational point of view, the tasks performed by Cataloga can be summarized as follows:

- automatic reading of a text;
- computing of all the occurring terminological compound words, i.e. location and computing of all the occurrences of any of a finite number of compound words;
- statistical computing of the ratio between terminological and non-terminological occurrences;
- statistics-based listing of all the terminological occurrences, in decreasing order and classed on the basis of the pertaining knowledge domains.

The technology used to assemble this software is “Borland Developer Studio 2006” (i.e. Delphi 10 or Delphi 2006). Delphi is a powerful RAD (Rapid Application Development) visual software development tool, based on an Object Programming Language. On the contrary, no specific hardware architecture is explicitly required, given the fact that the software is normally installed and used on both house-desktop and laptop standard Windows computers. During the start-up phase, considering a Windows XP computer equipped with 3 GB of RAM and a 1.66 GHz dual-core CPU, the software loads and preprocess a whole electronic dictionary (approximately 500,000 entries) in less than 10 seconds. This preprocessing step is performed only once for each software session. On the contrary, texts processing step is achieved in real time. The time complexity of Cataloga dictionary pre-processing algorithm is O(n), i.e. is linearly proportional to the sum of compound word lengths. At the same time, the matching algorithm has an O(m)-time complexity, i.e. it takes m<X<2m state transitions to process a text string of length m. All terminological compound words can be simultaneously recognized in one pass. During the analysis procedure, the terminological electronic dictionaries are completely allocated in RAM, so that the effective software execution time is very short and does not depend on the dictionary or the text size. The lingware embedded in Cataloga is developed also taking into consideration two specific formal and linguistic considerations:

- in any given scientific and/or technological text, the large part of cognition is conveyed by a small number of terminological words, which may be ontologically classified on the basis of the knowledge domain(s) in which they have been created and for which they express precise and non-ambiguous meanings;
- in most languages, there is a close and necessary relation between terminology and a specific subset of multiword expressions, i.e. the one formed by compound words[30]. This is proved by the fact that specialized lexica are mainly formed by compounds, in an average that is often higher than the 80% of the whole registered lexical set. Therefore it is possible to state that

[30] As already stated, within the complex set of multiword units, compound words have specific formal, morphological, grammatical and semantic characteristics which push towards a clear differentiation from other multiword units. Actually, the definition of compound words is based on two different aspects. The first is at the same time morphogrammatical and semantic, and refers to compounds as to lexical meaning units, i.e. sequences of words different from free groups or phrases; each compound word, and more specifically each compound noun, is in fact a single unit, with a precise grammatical function, and a specific meaning which cannot almost ever be inferred from the words that compose it. The second is essentially formal: it introduces a distinction between uninterrupted sequences of letters limited by blanks and forming simple words such as *sedia* (*chair*) and sequences of words separated by blanks or other diacritics elements, such as *sedia a dondolo* (*rocking chair*). For more specifications see paragraph 1 and 2.2.

> from a formal and semantic point of view, terminology fully exploits the syntagmatic procedures of compound word structuring and formation, in which a starting lexical element – for instance a noun with a generic meaning such as *carta* (*card*) – can be specified by adding other lexical elements, as in *carta di credito* (*credit card*, *debit card*).

In the Section 5 we will show how to use both software packages, NooJ and Cataloga, in a Hybrid Model of NLP, in order to transform fully oriented linguistic resources into effective and enhanced solutions for KMSs.

3 *Natural Language Formalization*

We have already seen why natural language formalization is to be achieved in order to reach different and complex NLP goals ranging from IR to parsing and machine-aided translation. Besides, apart from TGG, LG and statistic-based analyses, other formalization methods exist which model linguistic data to exploit them within NLP routines based on different theoretical, investigative and pragmatic purposes.

Actually, linguistic corpora have been annotated by means of SGML-based markup languages for almost 20 years. We can, very roughly, differentiate between three distinct evolutionary stages of markup technologies:

1. originally, single SGML tree-based document instances were deemed sufficient for the representation of linguistic structures;
2. linguists began to realize that alternatives and extensions to the traditional model were needed. Formalisms such as, for example, NITE were proposed: the NITE Object Model (NOM) consists of multi-rooted trees (Carletta *et al*., 2003; Evert *et al*., 2003);
3. we are now on the threshold of the third evolutionary stage: even NITE's very flexible approach is not suited for all linguistic purposes. As some structures, such as these, cannot be modelled by multi-rooted trees, an even more flexible approach is needed

> in order to provide a generic annotation format which is able to represent genuinely arbitrary linguistic data structures.

Therefore in this section, as for natural language formalization, we will present a state of the art focusing on the most relevant models of linguistic data structuring, enhanced KMS solutions and NLP application for KMSs.

3.1 *Models of Linguistic Data Structuring*

In this paragraph we will be briefly describe some models for linguistic data structuring, which mainly focus on different types of annotations and offer differtent solutions to language formalization.

3.1.1 *PAULA XML: Interchange Format for Linguistic Annotations*

PAULA XML is the *Potsdamer Austauschformat für linguistische Annotation* ("Potsdam Interchange Format for Linguistic Annotation"). PAULA XML has been developed in Project D1: *Linguistic Database: Annotation and Retrieval* of the SFB 632[31]. It is an XML-based standoff representation format, which has been designed to represent data with heterogeneous annotation layers produced by different tools. For visualization and querying of PAULA XML data, the database ANNIS can be used.

[31] For more information see http://www.sfb632.uni-potsdam.de/d1/paula/doc/.

Special features of PAULA XML are:

1. Based on a graph-based object model, the "PAULA Object Model"

   - Capable to represent any kind of linguistic annotation for textual data, in particular:

     ○ support for overlapping annotations, including conflicting hierarchies;

     ○ support for discontinuous constituents, including crossing branches (e.g., in syntax graphs).

2. Standoff-format specialized for multi-layer annotations with arbitrary linguistic annotations

   - Extensibility: It can be easily augmented by new layers.
   - Native support of structure-building relations (e.g., syntactic or discourse-structural dominance) and pointing relations (e.g., co-reference, or alignment)

3. Hierarchical organization

   - PAULA XML allows to group together different annotations attached to one single texts, multiple texts that stand in a specific (e.g., translation) relationship, subcorpora and corpora.
   - Metadata: PAULA XML supports metadata on the level of annotation layers, documents, subcorpora and corpora.

4. Technical infrastructure

   - (Partial) validation via DTDs and XML Schema.
   - Designated input format of the ANNIS database.

3.1.2 *EXMARaLDA*

EXMARaLDA defines a data model for the representation of spoken interaction with several participants and in different modalities. The data model is based on the annotation graph approach (Bird & Liberman, 1999), i.e., it departs from the assumption that the most important commonality between different transcription and annotation systems is the fact that all entities in the data set can be anchored to a timeline.

EXMARaLDA defines a basic version of the data model which is largely similar to other data models used with software for multimodal annotation (e.g., Praat, TASX, ELAN, ANVIL). This has proven an appropriate basis for the initial transcription process and simple data visualisation and query tasks. An extended data model that can be calculated automatically from the basic version by exploiting the regularities defined in transcription conventions caters for a more complex annotation and analysis.

Data conforming to this model is physically stored in XML files. Although the structure of the XML-files is given in a DTD, the graph model does not make use of XML's strength to formulate constraints on hierarchical relations and defining tag sets or annotation vocabularies.

Conversion filters have been developed for legacy data. Due to a lack of documentation and several inconsistencies in these older corpora, however, a complete conversion cannot be accomplished automatically, but requires a substantial amount of manual post-editing.

At the present time, linguistic data represented in the EXMARaLDA data format is usually created with the help of the EXMARaLDA Partitur-Editor, a tier-based tool presenting the transcription to the user as a musical score supporting the creation of links between the transcription and the underlying digitized audio or video recording. Alternatively, compatible tools like ELAN, Praat, or the TASX annotator can be used to create EXMARaLDA data. The EXMARaLDA corpus manager is a tool for bundling several transcriptions into corpora and for managing and querying corpus metadata. ZECKE, the prototype of a tool for querying EXMARaLDA corpora, is currently evaluated. The EXMARaLDA tools are described in detail in Schmidt & Wörner (2005) and in various materials available from the project website (http://www.rrz.uni-hamburg.de/exmaralda).

The transfer from the directed graph structure of transcription-graphs in EXMARaLDA to a data model which is hierarchy-oriented (e.g., single-rooted or multi-rooted trees) has to be accomplished via the graph's ordered nodes that establish the structure and are the only valid markers as to how annotations are linked to textual content. These nodes are translated into anchor points in the "root" – XML-file. The segments of the textual content link to their start and end anchors are maintained in a separate XML file.

Annotations on the textual content again link to these segments via pointers, so that the relations between the text and the annotations do not have to be calculated by means of the anchors.

3.1.3 *TUSNELDA*

Tusnelda is an acronym for the German translation of "Tübingen collection of reusable, empirical, linguistic data structures". This collection contains heterogeneous corpora that differ with respect to several aspects (e.g., annotated languages, text types, kind of annotated, language-related information).

Nonetheless a common annotation scheme, also called Tusnelda, has been developed several years ago. The development of the Tusnelda annotation scheme was heavily influenced by the work of the Text Encoding Initiative (TEI) and by the TEI-influenced Corpus Encoding Standard (XCES).

In contrast to the Exmaralda data format, Tusnelda does make use of a hierarchical data model, and all the Tusnelda corpora consist of XML-files which have been validated against the Tusnelda Document Type Definition.

The following example shows a Tusnelda file. The linguistic aspects of this extract of the Tibetean corpus can be found in Wagner and Zeisler (2004):

```
<clause>
<ntNode> <tok>
<orth>khra•phru•gu</orth>
<pos>NOM:anim~pers</pos> </tok>
<ntNodeCat>NP</ntNodeCat> <desc>
<case>Abs</case> </desc>
</ntNode> <tok id=”v6”> <orth
n=”2”>med-tshug</orth>
<pos>VFIN</pos> <desc> ...
<realFrame> <realComplement id=”v6c1”
status=”empty”> <role>POSS</role>
<ref target=”v5c1”> </ref>
</realComplement> <realComplement id=”v6c2”>
<role>EXST2</role>
</realComplement> </realFrame>
</desc> </tok>
<clauseCat>simple</clauseCat>
</clause>
```

This extract shows a standard XML-structure. However, a closer look reveals implicit information. The natural (and intended) way of interpreting the transcribed and annotated utterance is to relate the node "<pos>" to the node "<orth>", i.e., to relate a transcribed word with information on its part of speech. From an XML-oriented point of view, however, the nodes "<pos>" and "<orth>" are simply adjacent nodes. Another example of adjacent nodes are the first and the second token "<tok>". Hence, in the case of "<tok>" two neighbouring tags represent a sequence but in other cases (e.g., "<orth>" and "<pos>", or a sequence of <realComplement>) two adjacent tags provide different additional information with regard to the same text.

The general data format should avoid ambiguities of this kind. Of course, a general format without these ambiguities would lead to the necessity of transforming the Tusnelda corpora into the new format. Ideally, this transformation should be able to resolve the described ambiguities automatically.

3.2 *Enhanced Solutions for Knowledge Management Systems*

In this paragraph we will give some brief and necessary definitions of KM and KMSs. We will principally focus on their conceptual aims, main functions and on the range of possibilities they offer to manage and govern different types of knowledge, allowing the adoption of innovative solutions.

3.2.1 *Defining Knowledge Management and Knowledge Management System Structure*

As already stated, there is no universally agreed-upon definition of both KM and KMS, due to the fact that it is also very difficult to agree upon what knowledge essentially is constituted by. Therefore, to be coped with correctly, the concept of KM must be intended in the broadest way possible, particularly "as the process through which organizations generate value from their intellectual and knowledge-based assets". Most often, such generation of assets involves codifying specific amounts of knowledge possessed for instance by employees, partners or customers. Also, in most cases it involves the sharing of information among peoples and institutions.

From a general point of view, KM can be defined as a set of systems and actions suitable to store, disseminate, apply, refine, and create significant knowledge. Also, KM is based on the correct comprehension of all knowledge forms to be coped with; the best means to by which these forms of knowledge can be disseminated; how new forms of knowledge can be generated, acquired, learnt, and shared.

Basically, KM helps in considering knowledge as a tangible benefit. The main task in KM is to accurately and promptly transfer knowledge to those peoples who may need and reuse it. Even if this does not seem to be a complex task, it implies a deep understanding of the knowledge nature to transfer, and a consequent formalization of the whole transfer procedure. Furthermore, KM may alternatively focus on new knowledge creation, on pre-existing knowledge sharing, storage, and refinement, or on both.

On the other hand, a KMS can structure any kind of knowledge sets, ranging from the most simple to the most complex ones. For instance, the following example of KMS, taken from http://criticaltechnology.blogspot.com/2006/10/community-knowledge-management.html, describes “the resources and approach required to build a Community Knowledge Management System (CKMS) in rural developing communities.” (See Figure 3).

Its structure represents an outstanding example of enhanced solution for KMSs, considering that “The increased availability of Information and Communication Technology (ICT) through telecentres, cellular telephones, rural wireless networks and community schools have increased the likelihood of partnerships successfully creating community repositories of indigenous knowledge. Through the use of free open source software (FOSS), access to the multimedia of video recorders, audio recorders and digital photography combined with the increasing knowledge of how to use these technologies makes a CKMS within reach for many developing communities. Having the methods to gather, store, retrieve and distribute community knowledge through local partnerships and emerging ICT further reduces the knowledge divide”.

In any case, it is important to stress that technology is not KM in itself, even if it often facilitates KM. But from a more strict point of view, the previous figure demonstrates that KM and KNMs can be mapped and structured as ontology networks, and that each hub/node/oriented path of such ontology networks necessarily include linguistic expressions, the large part of which can be formalized and reused as basic part of NLP routines.

Generally speaking, a well-structured approach towards KM must create shareable values while also forcing, improving and refining knowledge assets to meet specific goals and targets. So, as far as the topics we are dealing with are concerned, KM implementation may have two different yet crucial dimensions, that is to say:

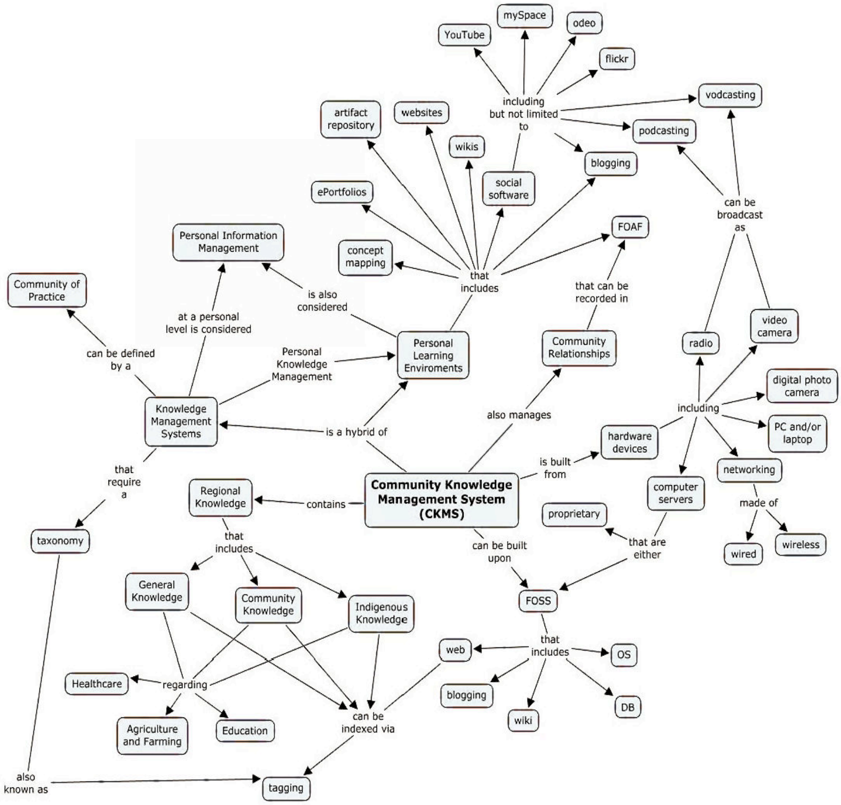


FIGURE 3. A concept map for a Community Knowledge Management System (CKMS) in rural developing communities[32].

[32] The figure is taken from http://criticaltechnology.blogspot.com/2006/10/community-knowledge-management.html.

- organizational (with reference to process correctness, environments, culture and systems);
- technological (with reference to the right and correctly implemented technological systems and tools to apply).

3.2.2 *Different Types of Knowledge*

Before starting to cope with KM formalization, it is necessary to distinguish between knowledge in itself, on one hand, and information or data, on the other. The difficulty of this task is essentially given by the fact that the word "knowledge" may have several different meanings even in specific fields, or also in one single discipline.

The word knowledge is often used in everyday language with reference to both theoretical and practical sets of information (for instance, both with reference to "wisdom" or to a specific know-how). From an ontological point of view, a large part of the complexity of knowledge definition comes from the necessity of interrelating its conceptual explanation to other concepts, and mainly to those concerning data and information, two terms which very often are considered as subordinate definitions of knowledge. But also other concepts like skills, understanding, and experience are widely used to define knowledge and KM.

At the same time, disciplines which are strongly oriented towards technology and greatly involve information systems very often cope with knowledge in the same way as with information, which is seen as a material easily codable and transmittable. So, today, some KMSs are essentially information management systems in which knowledge is virtually used as a synonym for information.

As for KM structure, Theirauf (1999) identifies three main components: unstructured data as the lowest point; structured data (i.e. information) as the following level, considering that data become information when are inserted into specific contexts, categories, and are submitted to calculus and condensation; and finally information about information, i.e. knowledge which is partly know-how, partly experience and partly understanding. According to Gamble and Blackwell (2001) "Knowledge is a fluid mix of framed experience, values, contextual information, expert insight, and grounded intuition that provides an environment and framework for evaluating and incorporating new experiences and information. It originates and is applied in the mind of the knowers. In organizations it often becomes embedded not only in documents or repositories, but also in organizational routines, practices and norms".

Also, knowledge may be classifiable as explicit and tacit. Explicit knowledge is codified and sometimes referred to as know-what (Brown & Duguid, 1998); as for KMSs, this is the type of knowledge which must be handled by means of linguistic formulas, and may be found in databases, memos, notes, documents, and so on (Botha *et al*., 2008). To extract explicit knowledge from documents and other records, as well as discovering knowledge within existing data and knowledge repositories, the main tools/practices in this case include Data Mining (DM) and Text Mining (TM).

On the contrary, tacit knowledge (also known as embodied knowledge) is not codified and/or based on personal/individual experience. Basically, tacit knowledge can be reconstructed by means of linked linguistic data and formulas, consequently through ontologizing concept maps as the one presented in Figure 3. Being a suprasegmental characteristic of human experiences, tacit knowledge is always present inside texts and documents, but as for what seen with explicit knowledge, its discovery and detection are more complex tasks, as this form of knowledge is never expressed by nor consequently is directly retrievable by means of formalized/iterative linguistic formulas. Consequently, concept ontologies are the most suitable mean to retrieve tacit knowledge and to account for it inside KMSs. As already stated, tools/practices useful in this process are all mainly based on observation and conceptualization.

According to Nonaka (1994) and Botha *et al*. (2008) tacit and explicit knowledge are always co-occurring and interacting; therefore, knowledge is almost always a combination of both elements. It is also worth noting that tacit knowledge slightly differs from embedded knowledge, which is constituted by processes, products, culture, routines, artefacts, or structures (Horvath, 2000; Gamble & Blackwell, 2001). Embedded knowledge (i.e. non explicit knowledge) may be found inside rules, processes, manuals, organizational culture, codes of conduct, ethics, products, and so on. Also in this case concept ontologies may be the most suitable mean to retrieve it and to account for it inside KMSs. This implies an examination and identification of the knowledge trapped inside organizational routines, processes, products etc., which has not already been made explicit. Management must essentially ask "why do we do something a certain way?". This type of knowledge discovery involves observation and analysis, and the use of reverse engineering and modeling tools.

3.2.3 *From Knowledge Management to Enhanced Knowledge Management Systems*

KM can be enhanced by means of frameworks and models, mainly in order to amalgamate different elements/concepts through structured ontological linguistic relationships. To achieve this task, basically the first thing to do is to:

1. identify the aims for which a specific knowledge management is to be built;
2. categorize the knowledge resources needed (i.e. all the concepts to ontologize);
3. identify (if present) the hierarchical links among the knowledge resources needed;
4. choose the most appropriate linguistic formulas to apply during the ontologization process (both arbitrary and/or non arbitrary, that is to say coming either from natural language, or from formal languages, or from both types);
5. validate and debug the KMS obtained so to retrieve structured and reusable knowledge from it.

As already stated, the role IT can play in a similar procedure is strictly connected to NLP automatic routines such as TM and DM. These routines, and also Content Management Systems (CMS), may be used to update, distribute, tag and deal with any form of linguistic matter. To achieve coherent results, the main function these routines must accomplish is the data importation, analysis, advanced indexing, searching, and retrieval. But at any rate, even if IT is a very helpful tool in explicit knowledge and information and management, humans' role is still crucial, especially in KM structuring and KMSs evaluation. In brief, this means that as for both KM and KMSs, only humans may assess appropriate implementation. Also, IT is fundamental for information management, but it is worth stressing that information in itself is not equal to knowledge. At the same time, it is definite that IT can be used to provide knowledge modeling and retrieval tools, by means of which to "read" and "interpret" ontology and conceptual mappings. IT may provide access to data and information, and IT systems can also be used to evidence trends in data and information. Finally, IT tools can also be employed in KM innovation process. In any case, we must stress that KM is not a technological discipline; it is more about managing people, culture, and organizational practices & structures. Only if IT is used right – as a supporting and enhancing mechanism for sound, existing KM practices – it can be a very valuable tool indeed. Effective KM initiatives are therefore never technology driven, and one should never seek a total KM "solution". In fact, it would must warn against any system that lays claims to that title. Doing so implies that either the developers have no issue promising far more than they can deliver, or they have no idea what a KM tool can and cannot do. Neither is a good scenario.

3.2.4 *Knowledge Management Systems*

Generally speaking, KMSs may be defined as IT systems in which to store and from which to retrieve knowledge, and also by means of which to improve and enhance KM processes. Even if this seems to be a somehow imprecise definition, it is worth remembering that today still there is no commonly agreed definition of what a KM or KMS can be. In any case, as for the topics here discussed, we can observe that one of the most important characteristics of KMSs is the fact that they may be based on ontologies maps, and also on FSAs/FSTs in which nodes and hubs can contain predefined or formalized linguistic matter. As we will see, this form of structuring would easily allow the successful application of NLP routines for structured knowledge retrieval.

Besides, James Robertson (2007) stresses that in themselves KMSs make no miracle, i.e. that even if they are formally well structured, what counts most is their (linguistic) content and the way they are processed coped with. So, as for KMSs a crucial aspect is the functionality of the IT systems that are required and/or planned to manage a specific kind of knowledge.

As for specific linguistic tools on which to found KMSs, we can indicate[33]:

- Data warehousing
- Data mining
- OLAP (Online Analytical Processing)
- Content management systems
- Document management systems
- Semantic and ontology networks

Data warehousing consists in creating linguistic databases in a central system (for instance a network server) to use as corpora during automatic textual analysis. In other words, data warehousing produces

[33] The list is adapted from Wickramasinghe, Gupta and Sharma (2005); see also Wickramasinghe *et al*. (2009).

large sets of different types of digitized texts, from which information and or knowledge may be automatically extracted. As we will see, best automatic routines to achieve this kind of extraction are those which combine lexicon-grammar automatic textual analysis and parsing, ontology-based analyses and statistical-based textual reading routines. Also, usually data-driven DSSs are based on data warehouse.

The following data warehouse model is taken from Thierauf (1999) and shows the process of warehousing data, extraction, and distribution (see Figure 4). As we can see, the NLP tools we are describing can be embedded in the boxes *Product application*s and *Data extraction*.

The figure implicitly tells us that the design and implementation of warehousing data is a crucial step in KMS structuring. Despite the necessity to build a functionally agile system, the main points to clearly and previously define are:

- the size of the database and the complexity of the analytical requirements;
- how users will receive the information;
- how routine decisions must be automated;
- finally, how users with different technical skills can access the data, considering that all the tools to apply will be based on the NLP routines which will be discussed later on.

According to Frank (2002), another critical aspect for the success of data warehouse functioning is the implementation of metadata, in other words the implementation of a linguistic system in which formalized data may say something clear about and automatically retrievable from unstructured data. In this sense, we can state that Frank (2002) straightforwardly but maybe involuntarily copes with ontologies and ontology building and exploitation.

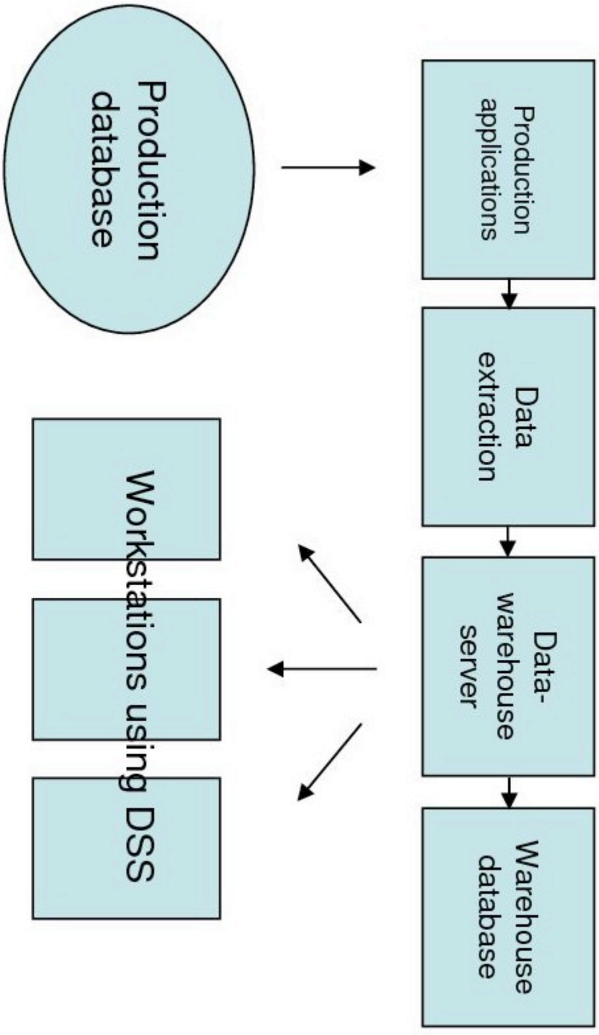


FIGURE 4. Data warehouse model[34].

[34] The figure is taken from Thierauf (1999).

A similar ontologization procedure is exposed in Parankusham & Madupu (2006), who describe the different roles metadata may have, that is to say data characterization and indexing, data access facilitation or restriction, data source determination and data currency. They also identify metadata lifecycle in term of collection (identification and capture), maintenance (metadata updating to match data architecture changes) and deployment (users access the relevant metadata, on the basis of their needs).

Furthermore, as for data warehousing structuring and implementation, on www.syntelinc.com the following five criteria are presented as crucial:

- *Recognize that the job is probably harder than you expect*. A large portion of the data in data warehouses is incorrect, missing, or input in such a way that it is not usable (e.g. historical databases that have not been updated to modern schemas).
- *Understand the data in your existing systems*. Analyze existing databases. Identify relationships between existing data systems so as to avoid inconsistencies when these are moved to the warehouse.
- *Be sure to recognize equivalent entities*. Identify equivalent entities in heterogeneous systems, which may appear under a different name.
- *Emphasize early wins to build support throughout the organization.*
- *Consider outsourcing your data warehouse development and maintenance*. Implementing a data warehouse can be a huge task that can often be better handled by experts. Many data warehousing applications are suited for outsourcing.

Therefore, the goal of a well structured and implemented data warehouse is to significantly reduce the time required to automatically extract knowledge. In order to achieve this task, other specific tools can be used, i.e. Online Analytical Processing System (OLAP), NLP procedures, and data visualization.

As for the topics here discussed, the most important features offered by OLAP are:

- query and reporting, i.e. the possibility to formulate queries using natural language instead of the programming language of a given database;
- statistical analysis, i.e. a function intended to transform large quantities of data into linguistic formulas helpful to produce answers to the queries,

So, it is possible to state that OLAP summarization of data and information may be used to structure coherent and effective knowledge extraction from structured and unstructured databases.

Semantic-Based DM is another process which can be used to extract knowledge or information from data warehousing. It is different from classical DM because it is dictionary-based and uses terminological lexical ontology to automatically extract meaning from texts. As it happens with classic DM, it may be used to minimize, filter, extract or transform even large corpora into summarized information. Semantic-Based DM also employs symbolic pattern description languages and statistical analysis, both to locate specific terminological compound words inside texts, and on the other hand to measure semantic characteristics, which are then divided into classes and clusters.

3.2.5 *KMSs and Data-Driven Decision Support Systems*

Data-driven DSSs are generally used to enhance decision-making process and help humans in problem solving procedures. They can access and manipulate data, they work with a data warehouse, use an OLAP, and employ DM techniques to retrieve knowledge from corpora. As for KM and KMS, data-driven DSSs can be important tools in increasing the range of crucial information retrievable by means of queries in natural language. Also CMSs can be useful tools to create, manage and distribute linguistic information on websites, as they allow the tagging of content with metadata (i.e. the tagging of linguistic content with keywords to facilitate data searching and retrieval).

As well, as components of CMSs, document manage systems may be used to publish, storage, index and retrieve documents. The mains functions of such systems are capturing (i.e. paper document scanning), classification by means of metadata (functional to ontologize documents conceptual contents using specific linguistic tags as keywords, dates, author names, and so on), tokenization and indexing (used by NooJ to portion files into lexical linguistic units), searching and retrieval (also present for instance in systems as NooJ and achievable by means of boolean rational expressions, concordances and formal/ local grammars). If compared to systems which are not based on automatic routines, document management systems offer an elevate speed of search, together with a greater precision and a higher efficiency with regard to knowledge retrieval.

3.3 *NLP Applications for Knowledge Management Systems*

Generally speaking, or better from a purely theoretical point of view, considering the way in which they operate plus their information structure and mission, Web search engines and crawlers should be considered as the most effective NLP applications for KMSs. However, anyone in need to use these engines, both in their basic and/or in advanced search functions, finds out that this is not true, and that almost always Web search engines are unable to carry out their main task, i.e. retrieving the information, data and knowledge for which they are queried. Actually, in this paper, it would be exceedingly long and unprofitable to address all the issues and highlight all the reasons why Web search engines and crawlers in practice do not accomplish the information tasks for which they were born. In the previous pages, we have several times had the occasion to state that the main reason to this *défaillance*[35] is a low sensitivity towards natural language, i.e. the fact of not having understood since from the outset that natural language has its own life, is mainly observable and describable as an empirical object and consequently must be primarily investigated as such. Each current limit of Web search engines is actually due to this miscalculation, the resolutions of which today seem too complex and too costly, in terms of time and also of economic and human resources.

So, in the following paragraphs, we will give account of three of the most important NLP applications for KMSs, that is to say Word-

[35] Also, a very clear and concise *résumé* of the main reasons why natural language queries do not work on Web engines is on http://inbentasemanticsearch.wordpress.com/2012/01/23/5-reasons-natural-language-search-might-not-work-for-your-company/.

net, FrameNet and Kim. We will mainly concentrate on their specific functions, and on their possible use inside linguistic-based knowledge extraction. In addition, we will highlight how these two systems have much developed their lingware parts, but have chosen to depend almost exclusively on Web site environments as regards the part relating to the software. For this reason, rather than real NLP applications, they can be defined as search engines having specific query functions for SW experimentations.

3.3.1 *WordNet*

WordNet® (http://wordnet.princeton.edu/) is a sizeable English[36] lexical database in which nouns, verbs and adjectives are subdivided into synsets, i.e. sets of cognitive synonyms. Each synset expresses a specific concept, while conceptual-semantic and lexical relations interconnect all synsets. Even if it looks like a thesaurus, WordNet may be used in computational linguistics and NLP.

WordNet has 117,000 synsets, all interconnected by means of conceptual relations. The interconnections established by WordNet exclusively concern specific words senses and are based on the labelling of the semantic relations existing among them. Each sysnset is composed by a gloss, i.e. a short definition of the concept on which it is based, and eventually one or more sample brief sentences to explain how to employ the words included in the synset.

[36] Up to today, English is the only language for which Wordnet descriptions are available. However, projects for all languages exist and/or are welcome.

Synonymy is the first relation looked for, i.e. the first synset established. Polysemous words are categorized inside all the synsets the conceptual definitions of which they take part to. In this sense, interconnections among WordNet word pairs are unambiguously structured. Other important relations among synsets are hyperonymy/hyponymy and meronymy.

Also verb synsets are hierarchically ranked; for instance, troponyms[37] are put towards the base of trees, and define progressively specific modes which typify events, as for instance communicate-talk-whisper. Modes expressed depend on pre-established semantic fields, as for instance “volume”, “speed”, “intensity of emotion”, and so on. On the contrary, directionally-oriented links are used to indicate how two verbs can necessarily and sillogistically include one another, as for instance buy-pay, succeed-try, show-see, and so on.

Antonymy governs adjective synsets, establishing pairs as wet-dry, young-old, fast-slow, which are direct antonyms with strong semantic bonds. These and other similar polar adjectives are interconnected on the base of their semantic similarity (i.e. “dry” to “parched”, “arid”, “dessicated” and “bone-dry”, or also “wet” to “soggy”, “waterlogged”, and so on). Adjectives having semantic similarity are defined as “indirect antonyms” with reference to the opposite pole member. At the same time, pertainyms are all those relational adjectives pointing to the nouns from which they are derived, as for instance “criminal” with reference to “crime”.

[37] A troponym is a verb which specifies more accurately the mode in which an action may be achieved by substituting a verb having a less specific meaning.

Finally, there are few adverbs in WordNet, mainly modal ones, directly managed as words derived from adjectives by means of suffixation.

WordNet mostly interrelates words belonging to the same POS. So, it is possible to state that it is actually formed by four sub-nets, i.e. nouns, verbs, adjectives and adverbs. Few cross-POS connections link words connoted by morpho-semantic invariance, as it happens with "observe-observant-observation-observatory". Also, for several noun-verb pairs, semantic role tags are used to specify functions as "location", with reference for instance to the pair "sleep-sleeping car", "agent" with reference for instance to the pair "paint-painter", and "result" with reference for instance to the pairs "paint-painting" and/or "paint-picture".

3.3.2 *FrameNet*

The Berkeley FrameNet project (https://framenet.icsi.berkeley.edu/fndrupal/about) is based on frame semantics and corpus analysis. It aims at enlarging and improving English on-line lexical resources, also accounting for word combinatory rules based on context study of semantic and syntactic features (i.e. valences calculated by means of senses that words acquire within sentence contexts). Sample sentences annotation is achieved by means of computer-assisted routines, together with tabulation and display of annotation results. The FrameNet lexical database so constituted actually includes more than 10,000 lexical units; of these, 6,000 ca. are fully annotated and ranked in about 800 semantic frames, linked in hierarchical connections, and illustrated by means of approximately 135,000 annotated sentences. Projects exist to create similar frame-semantic lexicons for other languages, and to work out automatic text labelling procedures based on semantic frame information.

FrameNet structures lexical units combining a word and a meaning within a specific frame. Polysemous words are included in all the required semantic frames, which are script-like conceptual structures describing particular types of situation, objects, or events, together with their actants and semantic roles. "For example, the 'Apply heat frame' describes a common situation involving a Cook, some Food, and a Heating Instrument, and is evoked by words such as bake, blanch, boil, broil, brown, simmer, steam, etc. We call these roles frame elements (FEs) and the frame-evoking words are lexical units in the 'Apply heat frame'. Some frames are more abstract, such as Change position on a scale, which is evoked by lexical units such as decline, decrease, gain, plummet, rise, etc., and has FEs such as Item, Attribute, Initial value and Final value. In the simplest case, the frame-evoking lexical unit is a verb and the FEs are its syntactic dependents: [Cook Matilde] fried [Food the catfish] [Heating instrument in a heavy iron skillet]"[38]. Frames are identified by predicating words (i.e. lexical entries) derived from these annotations. When heading specific structures, these predicating words categorize the given meaning of frames and the ways in which FEs are achieved.

In FrameNet, an annotation is achieved creating constellations of triples, i.e. given a sentence, interconnecting a specific FE (for instance, Food), a grammatical function (for instance, Object) and a phrase type (for instance, NP). Interconnections become then annotated layers to present in FrameNet annotation software. Annotation visualization is

[38] See https://framenet.icsi.berkeley.edu/fndrupal/the_book, page 5.

eased hiding both grammatical functions and phrase type layers. All layers and layer description are included in the data download set given on FrameNet homepage, together with complete frame and FE descriptions, frame-to-frame relations, and samples of lexical unit valence patterns.

With reference to WordNet and more generally to ontology building procedures, FrameNet structure is presented as having some specific and differentiating characteristics, and in particular the fact that:

- lexical units are provided with definitions taken from Oxford paper dictionary entries;
- multiple annotated sample sentences are given for each lexical unit and its senses;
- sample sentences are taken from concrete corpora and are not arbitrarily constructed;
- English lexicon analysis is achieved frame by frame rather than lemma after lemma; this helps in avoiding the use of the traditional alphabetic description/completion, which does not always support the correct explanation of word combinatorial and semantic characteristics;
- each lexical unit is not also linked to a given semantic frame, but also to all the other semantically similar words by which that frame is brought to mind;
- while WordNet and all ontologies are based on hierarchical relations between nodes, FrameNet uses a network of relations between frames, the most important of which are “inheritance”, “using”, “subframe”, and “perspective on”[39].

[39] For more on these relations, see https://framenet.icsi.berkeley.edu/fndrupal/the_book, chapter 6.

On the contrary, a drawback in FrameNet database is the fact that, unlike WordNet, it does not annotate nouns denoting artefacts and natural kinds. This may limit the ontologization process and reuse of the database in itself, reducing coverage of taxonomic hierarchical relations among (also terminologically describable) objects.

3.3.3 *KIM*

KIM (http://www.ontotext.com/kim) is platform for Knowledge and Information Management, the main functions of which are automatic semantic annotation, indexing, and retrieval of documents. It is based on GATE (General Architecture for Text Engineering)[40] and on a scaffolding allowing personalized and differentiated information extraction. The efficacy of the system, and consequently of all KIM-based applications, is granted by an upper-level ontology, built on a scheme of real-world entity concepts and knowledge, taken from massive knowledge bases, RDF(S) repositories (with compliance and possible extensions to OWL Lite), ontology middleware and reasoning.

In KIM, Semantic Annotation consists in automatically assigning to entities in given text links to pre-detected and pre-established semantic descriptions. In this way, automatic semantic annotation enables new routines, as highlighting, indexing and retrieval, categorization, generation of advanced metadata, smooth traversal connections between unstructured text and available relevant knowledge.

[40] For more on GATE, see http://gate.ac.uk.

KIM semantic annotation is applicable to any sort of text, including web pages, ordinary documents, and also database text fields. At the same time, knowledge acquisition is achievable also analyzing complex dependencies, as for instance relationships between entities, or event and situation descriptions. In such cases, automatic semantic annotation is definable as a traditional named-entity (NE) system, named-entity recognition (NER) and annotation process.

The KIM platform is composed by KIM Ontology (KIMO), a knowledge base, and a KIM Server with an application programme interface (API) for remote access, embedding, and integration. API also provides semantic annotation, indexing and retrieval services, and infrastructure. Documents are indexed by means of Lucene[41] IR engine, which allows cataloguing by entity types, measure relevance according to entities, together with tokens and stems.

[41] See http://jakarta.apache.org/lucene/.

4 *The Question of Linguistic Data Structuring Formal Models*

In the previous sections, we have already coped with the key concept of information flow management, stating that the more information is structured and classified according to strict data management standards, and more it is effectively manageable and exploitable. That is why in the field of linguistic knowledge bases, given a specific language, it becomes even more crucial to use a coherent and exhaustive formal description of its lexicon. And in order to develop well-working NLP applications, it becomes essential that such formal description is modularly reusable.

In this research, we want show how such a description should follow an empirical linguistic approach to NLP, based on the development of well-crafted LRs useful in the structuring of effective KMSs. This formalization must start from an accurate observation of linguistic phenomena, and from an appropriate linguistic data recording, in LR form, of all lexicon and lexical entry combinatory behaviours, encompassing syntax and, also, lexicon.

But retrospectively, we could remark that many theories of formal languages proceed from the need to provide a formal mathematical basis for such descriptions, as observed by Willem J.M. Levelt (2008) in the introduction of his book on grammars intended as formal systems.

The theory of formal languages originated in the study of natural languages, so the description of a natural language is called “Grammar”. A grammar indicate how the sentences of a language are composed of elements, how elements form larger units, and how these units are related within the context of the sentence. At this point, a preliminary clarification should be made: a formal language can serve as a mathematical model for a natural language, while a formal grammar can act as a model for linguistic theory.

In the mid-50s Noam Chomsky began to develop mathematical models for the description of natural languages. Two disciplines originated in his work and have grown to maturity. The first of these is the theory of formal grammars, a branch of mathematics which has proven to be of great interest to information and computer sciences. The second is Generative Transformational Linguistics (GTL)[42].

According to Levelt (2008) there are almost three types of formal systems. From a mathematical point of view, grammars are *formal systems*, like Turing machine[43], computer programs, prepositional logic,

[42] Chomsky’s GTL must not be confused with the transformational and distributional NLP approach structured by Maurice Gross on the basis of Zellig Harris’ formalisms, which we will deal with in 4.2. For more details on Harris’ studies referred to structural linguistics, discourse analysis and for the discovery of transformational structure in language see his bibliography, as indicated below in the references, and in particular 1946, 1951, 1968, 1970, 1991. Also, it is worth stressing that GTL mainly derives from Harris’ structural and transformational linguistics, even if Chomsky, as a former student of Harris, slowly detaches from the dictates of his teacher.

[43] A Turing machine is a device that manipulates symbols on a strip of tape according to a table of rules. Despite its simplicity, a Turing machine can be adapted to simulate the logic of any computer algorithm, and is particularly useful in explaining the functions of a CPU inside a computer.

The Turing machine was described by Alan Turing (1936), who called it an “a(utomatic)-machine”. The Turing machine is not intended as a practical computing technology, but rather as a hypothetical device representing a computing machine.

theories of inference, neural nets, and so forth. Formal systems characteristically transform a certain *input* into a particular *output* by means of completely explicit, mechanically applicable rules. Input and output are strings of symbols taken from a particular alphabet or vocabulary. For a formal grammar the input is an abstract *start symbol*; the output is a string of words, which constitutes a sentence of the formal language. Therefore a grammar may be considered as a *generative system*.

But according to Levelt (2008) there are also other two types of formal systems: *Automata* and *Grammatical Inference Procedures*. The first ones can be considered as *accepting systems*, i.e. systems which use the sentence of a language as input and, abstract symbol as output. The second type has sentences of a language as input and, as output, produce a grammar, which is in some way adequate for the language. Such systems serve as models not only for linguistic discovery procedure, but also for theories of language acquisition.

A different notion on formal models is introduced by Noam Chomsky who applies Harris' model of transformational analysis to Hebrew language, however soon deviating from it to develop a more abstract

Turing machines help computer scientists understand the limits of mechanical computation. Turing gave a succinct definition of the experiment in his essay, *Intelligent Machinery* (1948). Referring to his 1936 publication, Turing wrote that the Turing machine, here called a Logical Computing Machine, consisted of: «...an infinite memory capacity obtained in the form of an infinite tape marked out into squares, on each of which a symbol could be printed. At any moment there is one symbol in the machine; it is called the scanned symbol. The machine can alter the scanned symbol and its behavior is in part determined by that symbol, but the symbols on the tape elsewhere do not affect the behaviour of the machine. However, the tape can be moved back and forth through the machine, this being one of the elementary operations of the machine. Any symbol on the tape may therefore eventually have an innings». For more debates about the topic see http://plato.stanford.edu/entries/turing-machine/ and also an influential book of Hofstadter (1999) about the Church-Turing Thesis.

notion of transformation. Chomsky tries to resume a traditional grammar norm, that is to say the one which divides a proposition into Subject and Predicate. Also, he offers a sort of derivation of the sentences, as they came to surface coming from a *deep* structure, through transformations that bring into play rewrite rules deriving from the algebraic graph theory.

In 1957 he publishes a small book (*Syntactic Structures*) in which his interpretation of transformation as a link between a deep structure and surface combines with the harrisian idea about the existence of elementary sentences (at the level of simple sentences). So, Chomsky describes a project about syntax in which a few basic principles are taken:

- the linguist's task is to describe the mental linguistic competence of an ideal speaker-hearer (i.e. *competence*);
- concrete linguistic realizations, with all their idiosyncratic variability, do not constitute the primary topic of language (i.e. *performance*);
- linguistic competence can be described using a combinatorial mathematical model that takes into account the sentence context in which word combinations are realized (by approximation, Chomsky comes to the definition of a context-sensitive formal grammar, after demonstrating the limits of a formal grammar based on finite automata and of context-free formal grammars);
- this specific grammar formal model must be able to assign an appropriate phrasal structure to all sentences that respect the rules of a given natural language (this procedure is called *generative* in the sense that the grammar must generate only grammatical sentences of a language, and nothing else);
- syntax is to be considered a study of word combinations (to which is assigned the status of grammaticality or of *good form*)

sharply distinguished from the study of meaning (the semantic component), of which it is said that acts only at a later time, as a semantic interpretation (the same reasoning is applied to the phonological component);

- syntax uses an algebraic rewriting system that has a tree representation in which categorical symbols (noun phrase, name, verb phrase, verb, ...) act at the highest level, while concrete lexical items (terminals), which are part of the vocabulary, i.e. of the grammar lexicon, act at the lower level;
- transformation are seen as the tool allowing to account for word combinatorial realizations that surface analysis, such as the one in immediate constituents, cannot interpret as correlations or equivalences with other word combinatorial realizations;
- importance is given to the identification of elementary (or nuclear) sentence structures, from which by transformation also complex sentences can be derived.

Chomsky's project of 1957 produces a plethora of transformational studies aiming to describe English and other languages. This project is still sufficiently compatible with the model developed by Harris.

Especially in Anglo-Saxon countries, but with interesting developments in Europe, between 1957 and 1964 we see a rich debate, both on the generative-transformational model, and on the relationship between syntactic, semantic and lexical components (as an inventory of terminal elements).

With *Aspects of the Theory of Syntax* (1965) Chomsky redefines the program of TGG, outlining a model that will be named "Extended Standard Theory", and which sets out the new theoretical and methodological principles that it intends to pursue.

They can be summarized as follows:

- definition of lexicon as a list of entries characterized by selection features (animate, inanimate, etc.) and in particular, as regards verbs, as a list of sub-categorizations, that is to say requests or selections of specific lexical features for all further necessary complements;
- clear separation between deep structure and surface structure;
- consolidation of the transformational component which belongs to the core of syntax;
- clear separation between syntactic component (generative), and semantic and phonological components (interpretation);
- abandonment of the need to identify the elementary or nuclear sentence structures.

In *Aspects*, Chomsky takes a radical position against the attempts to introduce a generative semantics, affirms the primacy of syntax and expresses the idea of being able to account for lexicon irregularities by means of syntactic-transformational rules.

4.1 "*On the Failure of Generative Grammar*"

Soon after Chomsky's innovating theories, another concept of formalism was introduced by the French linguist Maurice Gross. Initially, Gross worked on the definition of the mathematical properties inherent in formal grammars and usable to describe tongues (1963; 1964; 1968; 1970; 1972; 1973). He started to consider transformations as a powerful tool to discover new facts about natural language, a kind of particle accelerator that allows a linguist to identify mechanisms, elements and data which are not directly observable.

By that time, the methods of transformational (generative) grammar had been available for more than twenty years. It was believed that, thanks to them, syntax had become a natural science. It was convincingly demonstrated at an early date that transformational models imposed on the description a precision and a coherence never reached before. But, as affirmed by Maurice Gross in his famous paper of 1979 *On the failure of Generative Grammar*, one may wonder why no linguist had been able to construct a transformational grammar with the type of coverage that traditional grammars used to provide.

Gross reached his conclusions after attempting to construct a TGG of French. «I and my co-workers – he affirmed – have built a formal grammar encompassing a significant portion of French, but we were unable to accomplish this without considerably modifying the theoretical framework. This grammar contains about 600 rules and conditions of application (we do not distinguish these two notions). We attempted to verify systematically the applicability of these rules to more than 12,000 lexical items. We were forced to conclude that we could obtain no generalization without a reasonably complete study of the lexical items of the language and their syntactic uses».

The aim was to build a classification for the data collected according to Generative Grammar (GG) and following its rules. After more than ten years of investigation, problems had become more and more significant, raised by these large-scale experiments and by theoretical elaboration of the resulting data.

GG could have been demonstrated to be a descriptive method far superior to all previous traditional and structural attempts. But the insistence on an experimental paradigm which depended entirely on introspection to provide the linguistic examples, and which was explicitly motivated by a desire to treat linguistics at an abstract level of argumentation, had caused the field to evolve toward some surprising philosophical speculations.

In 1968, Gross had started to describe the 3,000 French verbs that select a completive sentence, applying the generative model developed by Rosenbaum in 1967. He completed a first version in 1975, using an electronic data-base. As the work went on, he moved away from Chomsky's paradigm, and entered into open conflict with the TGG settings.

What had happened? The 3,000 French verbs he had analyzed reacted incoherently to the assumptions made by Rosenbaum (1967) for English: exceptions were more numerous than rules. When the properties analyzed for each verb exceeded the number of five or six, the classifications made showed that every verb had its own individual behaviour, almost completely independent from those of other verbs.

However, identifying at least one definitional property for a single important class of verbs, it was still possible to build a classification composed of about fifty classes. The only concrete problem was – and is – all in the evolving nature of the classification, or at least of its crucial part. The discovery of new phenomena, the revision of certain assumptions, lead to an update of the classes and to a recurrent maintenance of the entire classification.

Following Harris, Gross states that grammar already includes and is not separated from semantics, and that specifications on semantic statements are concretely possible only if they are based on non-metalinguistic analyses of natural language. Also, Gross discovers that it is doomed to failure any generalization made without a rigorous effort towards classification, verification and/or falsification of the initial hypothesis; and that in all languages, mainly with reference to the concept of transformation, syntactic structures cannot be separated from the concrete and unpredictable behaviour of single lexical units.

Also, in his 1979 paper, Gross states that theoretical arguments and problems are quite different in the two formal approaches, that of Chomsky's generative framework and that of Harris' algebraic system. Actually, Chomsky has attempted to construct a geometry for the deformations of trees, and his main purpose seemed to be a search for abstract conditions on the deformations. On the contrary, Harris had minimized the amount of formalization needed to relate sentences to each other, and had defined an algebraic structure on classes of sentences practically independent of the geometry of sentences.

While most linguists deeply believe that a grammar must be a formal system, Gross considers that the validity of the notion of geometry for constituent structure has not yet been demonstrated or even made plausible.

He continues pointing out that «[…] linguists have acquired a degree of snobbery that leads them to prefer handling a prestige vocabulary to painstaking experimental work. Brilliant dissertations, sprinkled with decorative symbols and equations, can be composed on such deep themes as a determination of theoretical and empirical conditions that should be met by Universal Grammar. Meanwhile, the ingenuity and concentration of efforts necessary to classify large numbers of structures do not lend themselves to the practices developed by pure theoreticians. Concrete effects of this attitude are visible. Normally, a specialist who invents some abstract mechanism should propose some way to verify its adequacy, or verify it himself; this can and should be done by applying the mechanism to all relevant parts of well-studied languages. […] This elementary rule is almost never followed. The justification of this system is supposed to be identical to the division found in physics between theoretical and applied or experimental research. To the extent that this view is meaningful, it might be justified by the enormous dimensions of the domain, but it is in no way thinkable for a field as narrow as English syntax or as ephemeral as trace theory; it takes only a few hours to extract from a dictionary the verbs that have no passive. An experimental scientist is perfectly willing to spend a few weeks or more at such an elementary but essential task».

Such severe comments from Maurice Gross allow us to be aware of the pitfalls that even the most complete theory and the most comprehensive system of language formalization can fall into while describing linguistic phenomena. Above all, such pitfalls become clear when the description of these phenomena claims to be universal and reproducible.

Indeed, the next paragraphs will highlight some "heuristic" solutions that partly solve such problems. They take into consideration not only "vagaries" of language, but also different sets of "exceptions" and co-occurrences in the combinatory use of linguistic elements. As we will see, thanks to its methodological basis, LG framework can support and manage formal linguistic models which have these characteristics.

4.2 *Lexicon-Grammar: a Theoretical and Methodological Challenge in the Formal Modelling of Linguistic Data*

As already stated, LG is a NLP framework which produces formal descriptions in which lexicon and syntax are considered as inseparable. Building a new system of language formalization is the main purpose for which Maurice Gross creates LG; he realizes that all previous descriptive methods and analyses achieved with traditional tools (i.e. descriptions based on TGG) are not reliable enough. For this reason, Gross borrows Harris' concept of transformation and makes of it one of LG methodological pillars.

Harris discovered transformations while he was developing a syntax theory based on more general terms. Up to that moment, words where usually classified inside classes, but no method existed to analyze word combinations. Moreover, Harris arrived to observe that sequences of word classes could lead to the identification of sentences subsets having comparable formal aspects. With reference to specific sets of sentences, Harris then started to map the preservation of precise properties from one subset to another, applying the same evidence method used for linear algebra transformations. So the term "transformation" began to be used also in linguistic studies, especially syntactic ones. In this way, Harris showed that starting from words combinatorial predispositions it was possible to recursively define specific subclasses having similar semantic features. Entire sequences of morphemes and phrases forming sentences were put into correspondence. Active and passive sentences were analyzed as being in a relationship of reciprocal transformation. Therefore, the step was short from this basic assumption to the identification of word categories (i.e. verbs) which determined the functioning of a complete sentence and ruled the saturation of complements. On such basis, Harris also identified the existence of elementary (or nuclear) structures of sentences, consisting of operators (i.e. verbs) and arguments (complements)[44]. In LG framework, Maurice Gross gave

[44] In Europe, somehow sheltered from the great currents of American and post-saussurean linguistics, also Lucien Tesnière assigns a decisive role to verb regency inside sentences. Tesnière (1953; 1959) has also introduced a new terminology, actually not always accepted, where instead of *regency* (fr. *rection*, ingl. *governement*) we find

concrete form to all these methodological passages, classifying verbal predicates on the basis of their distributional and transformational likeness (i.e. adopting Harris' mapping procedure) and using binary matrix tables to define sets of verbal predicates having similar formal and semantic features. These are the real reasons why Chomsky's position[45] and the one of Gross are today irreconcilable. According to Gross, the rules introduced by the TGG become only exceptions if one broadens the field of investigation. And if exceptions are so numerous that they cannot be statistically defined as such, then probably our linguistic competence is not as innate as it is stated by the Extended Standard Theory model: we rather acquire a large part of it in the course of our lives. Gross suggests that if this is what really happens, then it is necessary to rethink in a completely new way the role played by memory in the acquisition and production of the syntax of a language.

The challenge of Maurice Gross about language formalization and its algorithmic processing is clearly expressed in the following quotation from the paper *Lexicon-grammar and the syntactic analysis of*

value (fr. *valence*, ingl. *valency*). As for the concept of the required complements of verbs – a concept far from being clear in his time, and which is not yet completely clear – Tesnière proposed the term *actant* (fr. *actant*). The success albeit not immediate of Tensnière's theory has given a strong impetus not only to theoretical studies, but also the creation of systematic verb valency lexica. This occurred especially in Germany and with reference to the German. The *Valenzbibliographie* of Helmut Schumacher (1988) includes 2377 titles relating to 23 different languages and 41 language pairs contrastively examined. Their number has certainly much grown up in the meantime.

[45] From the late '60s to present days, Chomsky has gradually simplified the *Aspects* model, achieving the current one which is known as *minimalist* (Graffi 2001); after reducing all transformations to a single one (MOVE, i.e. displace), this model makes less crucial the separation between deep and surface structure. Chomsky's paradigm is now oriented towards the definition of an innate universal grammar, which in the act of forming itself is projected inside concrete languages, through a series of *parameters* specific to each one of them.

French (1984): «A lexicon-grammar is constituted of the elementary sentences of a language. Instead of considering words as basic syntactic units to which grammatical information is attached, we use simple sentences (subject-verb-objects) as dictionary entries. Hence, a full dictionary item is a simple sentence with a description of the corresponding distributional and transformational properties. The systematic study of French has led to an organization of its lexicon-grammar based on three main components: the lexicon-grammar of free sentences, that is, of sentences whose verb imposes selectional restrictions on its subject and complements (e.g. *to fall, to eat, to watch*); the LG of frozen or idiomatic expressions (e.g. *N takes N into account, N raises a question*; the LG of support verbs. These verbs do not have the common selectional restrictions, but more complex dependencies between subject and complement (e.g. *to have, to make* in *N has an impact on N, N makes a certain impression on N*). These three components interact in specific ways. We present the structure of the lexicon-grammar built for French and we discuss its algorithmic implications for parsing.»

It is clear that the LG model, inspired by the experimental sciences and based on Harris distribuzionalism, has formalization as its primary need, because it starts from exhaustive data collection (i.e. linguistic facts), comparing such data with all possible language uses, in order to achieve qualitative and quantitative analyses. Starting from *simple sentence*[46], i.e. the *minimal meaning unit* in which "selectional restriction rules" and "distributional" ones are applied, LG draws up a list of all possible transformational properties for each type of simple sentence. Syntactic-semantic properties are defined with the accuracy necessary to compare them systematically with all the entries of a lexicon. This

[46] For a more detailed definition of *simple sentence* see Gross (1988).

last step establishes the inseparable relationship between lexicon and syntax.

This linguistic model of formalization is becoming more functional to modern technological implications of computational linguistics. Thanks to corpus linguistics, for example, the systematic comparison of lexical entries and syntactic-semantic properties assumes large scale proportions and at a much smaller time. As well, ontological projects as FrameNet (Baker, Fillmore & Cronin, 2003) and VerbNet (Kipper-Schuler *et al*., 2006) evidenced, moreover, a convergence towards goals similar to those of LG.

In the next two paragraphs, we will discuss the importance of two approaches which are frequently used in linguistics and which support formalization and construction of language descriptive models. The first one takes advantage from statistical techniques and allows faster data processing. The second one concerns ontologies that, due to their classification power, can be a useful tool to support and manage language descriptive models in a more friendly way.

4.3 *Statistical Models: Faster Methods of Data Processing*

Statistical models can be a useful support in natural language analysis because they can provide fast data processing methods. This advantage is even more evident if we think of those language formalization models in which data are in a first step massively collected, and successively structured and classified for analysis, according to specific criteria. In some cases, statistics comes into play to facilitate those linguistic analysis tasks aiming at synthesizing many unstructured data into structured, manageable and reusable sets. However, these valuable statistical techniques must be handled with care in order not to invalidate their potentialities. It may happen that “processing speed” and “immediate availability of results” are incorrectly used to conduct analyses without taking into account solid theoretical bases and strictly linguistic methods. Statistics alone can lead to significantly inaccurate results; to avoid this, it must be necessarily accompanied by and hybridized with a full and exhaustive linguistic analysis of the object to analyze.

The origins of the area defined as textual statistics and TM must be found in the works of G.K. Zipf (1935) and G.U. Yule (1944), who can be considered as the main precursors both of modern linguistic quantitative analysis, and of its properties and applications in statistics. The same J.P. Benzécri (1963) based his first experiments of what will be *analyse des données* (1973, 1982) on the study of linguistic data (1981), in opposition to Chomsky’s ideas[47] and following Harris[48], whose late formalization[49] of linguistic structures aims at the individuation of immediate constituents by means of sentence segmentation; this formali-

[47] Chomsky argued that linguistics can not be inductive in the sense that grammar can not be deduced from rules found in a set of texts (corpus), but it can be only deductive, and that only starting from axioms it generates patterns of specific languages (Benzécri, 1982).

[48] In *Elementary transformations* (1964), Harris calls *word distribution* all its possible local contexts. In *Mathematical structures of language* (1968), he argues that discourse lends itself to a distributional analysis regardless of meaning, and he proposes to determine combination rules of the language in order to reveal the elementary relationships between different classes of concepts in a corpus. To this aim it is necessary to integrate to quantitative treatment of the corpus a morpho-syntactic analysis of textual data, i.e. introducing description algorithms of phrases that allow you to segment sentences of the text in their syntagmatic constituents, then, finally to identify them and clarify their internal relationships (Martinez, 2003, p. 275).

[49] See Harris (1976).

zation is very similar to the statistical approach to NLP. In the following, we will explore some statistical tools used for TM; such tools are referred to as a set of techniques that can link up well with strong and in-depth language analysis. Some of them are useful in NLP to treat data arrays resulting from lexical and textual analyzes performed on unstructured data corpora. In particular, as for visualization methods, we will discuss Factor Analysis, Latent Semantic Analysis/Singular Value Decomposition (LSA/SVD), Multidimensional Scaling; as for automatic classification methods, with text clustering, fuzzy methods; and also with some processes used to evaluate representations, as for instance bootstrapping.

4.3.1 *Statistical Analysis Tools and Procedures*

In order to cope with natural language ambiguity, statistics uses complex analysis of large textual data arrays applying methods and techniques taken from multidimensional analysis (correspondence analysis, cluster analysis, discriminant analysis, and multidimensional scaling). Such kind of analyses, which measure similarity of lexical profiles, produce contextual representations of textual information; these become "views"[50] in which, as for lexical units that can grasp the meaning inside investigated corpus, it is possible to apply the Gestalt principle "closeness vs. similarity" (Bolasco, 2005).

[50] The term "view" is here used to indicate the graphical presentation of the results coming from visualisation techniques used in statistical textual analyses.

As well, factor analysis is a statistical method used to describe variability among observed and correlated variables. Here, variability can be interpreted as the potentially lower number of unobserved and uncorrelated variables, called factors. In other words, it is possible, for example, that variations in three or four observed variables mainly reflect the variations in fewer such unobserved variables. Factor analysis checks for such joint variations to compensate for unobserved latent variables, and in some cases, it reconstructs the hidden or modal phrases (Bolasco, 1999) used as meaning models of texts. The observed variables are modelled as linear combinations of the potential factors, plus "error" terms. The information gained about the interdependencies between observed variables can be later used to reduce the set of variables in a dataset. Computationally, this technique is equivalent to low rank approximation of the matrix of the observed variables. Latent variable models, including factor analysis, use regression modelling techniques to test hypotheses producing error terms, while Principal Component Analysis[51] (PCA) is a descriptive statistical technique.

[51] "These techniques are typically used to analyze groups of correlated variables representing one or more common domains; for example, indicators of socioeconomic status, job satisfaction, health, self-esteem, political attitudes or family values. Principal components analysis is used to find optimal ways of combining variables into a small number of subsets, while factor analysis may be used to identify the structure underlying such variables and to estimate scores to measure latent factors themselves. The main applications of these techniques can be found in the analysis of multiple indicators, measurement and validation of complex constructs, index and scale construction, and data reduction. These approaches are particularly useful in situations where the dimensionality of data and its structural composition are not well known. When an investigator has a set of hypotheses that form the conceptual basis for her/his factor analysis, the investigator performs a confirmatory, or hypothesis testing, factor analysis. In contrast, when there are no guiding hypotheses, when the question is simply what are the underlying factors the investigator conducts an exploratory factor analysis. The factors in factor analysis are conceptualized as "real

Therefore and more specifically, Latent Semantic Indexing (LSI) is an indexing and retrieval method based on SVD, a mathematical technique which identifies relationships patterns among terms and concepts in unstructured corpora. LSI is based on the observation that words used in the same contexts tend to have similar meanings. A key feature of LSI is its ability to extract the conceptual content of a body of text by establishing associations between terms occurring in similar contexts.

As for automatic classification tasks, also texts clustering algorithms are widely used in statistical linguistics. Clustering is used to partition an unstructured set of objects into concrete clusters (groups). While applying most of clustering algorithms, two components are necessary: an object representation and a similarity (or distance) measure between objects. Also, it is necessary to distinguish between clustering and categorization, especially because categorization, within the meaning described here, lets a machine decide to which of a set of predefined categories a text belongs. On the contrary, in clustering, the machine decides how a given text set should be partitioned. Categorization is appropriate to categorize new texts according to an already existing categorization, clustering when it is necessary to discover new structures not already known. Both methods may give interesting results on unknown text sets; categorization sorts them according to a well known structure, clustering displays the structure of the particular set.

world" entities such as depression, anxiety, and disturbed thought. This is in contrast to PCA, where the components are simply geometrical abstractions that may not map easily onto real world phenomena. Another difference between the two approaches has to do with the variance that is analyzed. In PCA, all of the observed variance is analyzed, while in factor analysis it is only the shared variances that is analyzed." (Available on http://psych.wisc.edu/henriques/pca.html).

Basic clustering algorithms[52] may be defined as *hierarchical*, when they produce a hierarchy of clusters; *partitioning*, when they gives a flat partition of a given set; *hard,* when each object they locate belongs to only one cluster; *fuzzy* when the objects located belong to more than one cluster (usually with a degree of membership).

As previously mentioned, many of the statistical methods applied to word-by-document matrixes are closely connected. Text clustering may be used for dimension reduction in the same way as LSA; the cluster centroids may serve as a basis onto which the texts can be projected. This method gives similar results as LSA, but is more computationally efficient (Dhillon & Modha, 2001).

Finally, as indicated by Harald Baayen (2008), *bootstrap* is a technique which provides consistent means for validating cluster analyses. The basic idea underneath bootstrapping adoption in statistics is to assign accuracy measures to all the sample estimates. As regards the properties of an estimator, it is the practice of making estimation (such as variance) by measuring properties when sampling from an approximating distribution; for this last, one standard choice is the empirical distribution of the observed data. If a set of observations may be assumed to come from an independent and identically distributed population, such set can be implemented by constructing a number of resamples of the observed dataset (and of equal size to the observed dataset), each of which is obtained by random sampling with replacement from the original dataset.

Other most profitable methods of validation are *accuracy* and *precision*. Accuracy of a measurement system is the degree of closeness of measurements of a quantity to that quantity's actual (true) value. *Pre-*

[52] For amore detailed dissertation on clustering algorithms, see Jain et al. (1999).

cision of a measurement system, i.e. *reproducibility* or *repeatability*, which is the degree to which repeated measurements under unchanged conditions show the same results.

4.4 *Ontology-Based Models: a Survey on Classification Tools*

In recent times, ontologies have proved a valuable and useful data classification tool, mainly to underpin linguistic routines for knowledge modeling. Compared to other more simplified classification tools to network knowledge giving not only the opportunity to build relationships and links between data, but also, through logical descriptions used to construct ontologies, to achieve further inferences not directly evident in collected and classified data. Ontology-based models could present descriptive limits, especially if they are built in spite of syntactic-semantic criteria (i.e. semantic tagging) concerning the data collected. As previously seen with statistics, to overcome such descriptive limitations, also ontologies can be hybridized with extensive language investigations.

There are, indeed, a lot of works on modelling lexical information for ontologies, but not always all of those stick to deep linguistic criteria to model syntactic and semantic features. SKOS (Miles & Brickley, 2005), for instance, is a standard language used to represent lexical information. However, it presents the conceptual limitation of mixing up linguistic and semantic knowledge. In fact, to express semantic relations, SKOS uses two specific tags, *skos:broader* and *skos:narrower*, anyway not clearly and intentionally stating the semantics of these relations. Of no help are also two further definitions, i.e. the subproperties *skos:broaderGeneric* and *skos:narrowerGeneric*, used to have class subsumption semantics (i.e., they inherit the *rdfs:subClassOf* semantics from RDFS).

A useful attempt of making meaning explicit with respect to ontologies is given in Cimiano *et al*. (2007); a model is presented to equip classes with information about lexical realization, and in particular properties of a given domain ontology. Clearly, this approach aims at specifying how to map subcategorization frames in a way more functional to complex ontological structuring. Any NLP application in a specific domain have need of producing output structures compliant with a specific domain ontology. So, it is necessary to design a rich lexicon model which allows a declarative representation of the mapping between language and a given domain ontology. In such cases, RDF(S) vocabulary (Brickley & Guha, 2002) will not be suit the case, i.e. a model is needed which allows the definition of more complex structures, as for instance the lexicalization of concepts and properties.

In the lexicon ontology modelled by Cimiano *et al*. (2007), *subcategorization frames* represent linguistic predicate-argument structure. The following Figure 5, taken from the already mentioned paper, illustrates an ontology structure including a frame in which three essential parts of speech are specified: verbs, nouns and adjectives, each one described by means of linguistic features. This model represents a sound example on how to couple linguistic annotation and ontology design.

Therefore, in this model it is possible to increment the number of linguistic features useful to represent logical relations and properties of classified concepts in any considered domain ontology. For instance, if we want to add features as *semantic roles*[53], which are properties usable to describe lemmas in form of logical relations between concepts.

[53] As elsewhere stated, a *semantic role* is the logical and meaning relationship

It is possible to set up macro-intuitive semantic classes which correspond to specific sequences occurring inside texts and are therefore automatically recognizable.

At any rate, as for linguistically incrementable ontologies, two relevant observations can be made: first of all, semantic definitions and related ontologies essentially concern proper and/or common names; secondly, semantics is intended as a formal logical apparatus usable to hierarchically organize the conceptual system of classes, sub-classes and properties.

But it has been demonstrated (Elia, Vietri, Postiglione, Monteleone, Marano 2010) that within concrete sentences all these names only play the role of arguments, the predicates of which are external to the world of words, as they are represented by logic operators meant to connect, to infer, and to relate arguments. In this sense, it is possible to distinguish between *lexical ontologies*, which are based on characteristics useful to connote and/or denote words by means of lexical tags; and *syntactic ontologies*, which are based on characteristics allowing the definition of combinatory properties inside specific simple sentences. This means that, hierarchically and syntactically speaking, we have to formalise predicate properties before creating ontologies for proper and common names.

To better explain this necessary step, we will now focus on those semantic predicates that express the intuitive notion of “exchange”, i.e. “Transfer Predicates” (T) which require three arguments: a “giver” (D, standing for “datore” in Italian), an “object” (O) and a “receiver” (R). This relation can be formalized as T (D,O,R).

between an argument and its predicate; the notion of *semantic predicate* can be formalised on the basis of widely agreed upon logical procedures. For more on these topics, see Gross (1981), EMDA (1981) and Vietri (2004).

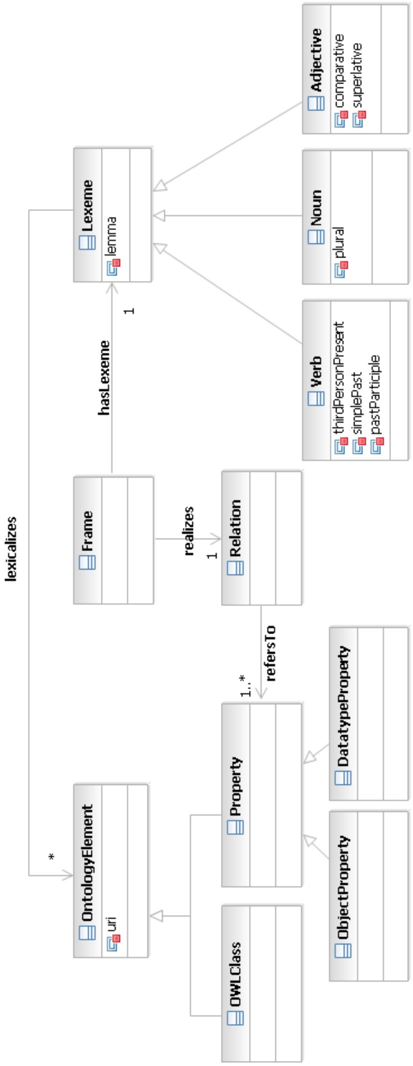


FIGURE 5. Example of linguistic features insertion inside ontology design [54].

[54] This figure is taken from the newsworthy research work of Cimiano *et al.* (2007).

Be *Sy* the set of lexical-syntactic forms of a language and *Se* the set of meaningful elements. Then, the elements of *Sy* can be associated to the elements of *Se* by means of the following interpretation rules (*R*):

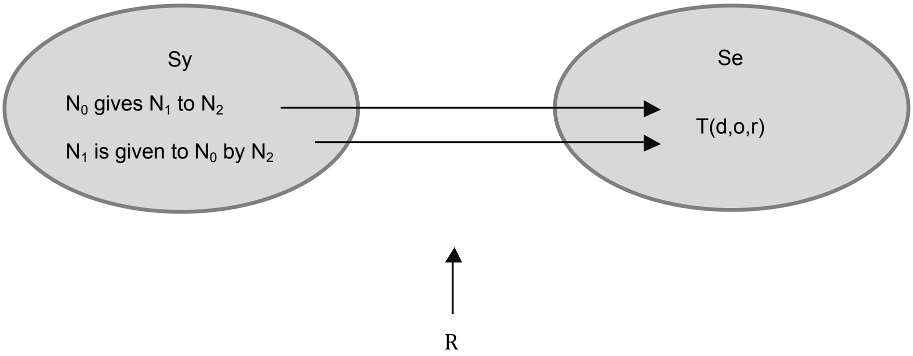


FIGURE 6. Representation of an active simple sentence of a Transfer Predicate "to give".

In the scheme above, the active simple sentence built on the verb form *dare* (to give, N_0 *V* N_1 *a* N_2) and the corresponding passive sentence N_1 *essere Vpp da* N_0 *a* N_2 (to be given by) are both associated to a unique semantic predicate T(D,O,R), according to an interpretation rule that associates N_0 to "D", N_1 to "O" and N_2 to "R". These predicates represent a particular case of a wider intuitive notion that denotes the transfer of an object (animate or inanimate) from one place to another.

5 *Hybrid Model of NLP*

In the previous section we have seen some formal models for linguistic data structuring. Since each one of them may present limitations in the accuracy of the data (i.e. Statistical Models), in the constrained generalization of the linguistic description at the expense of more particular and detailed phenomena (i.e. TGG Models), in the massive proliferation of data that issue from fine-grained analyzes (i.e. LG Models), in the loss of linguistic information (i.e. Ontology-Based Models), we propose below an Hybrid Model to use in NLP applications to create effective enhanced solutions for KMSs. The elements of this hybrid model are:

- a solid theoretical approach consisting mainly of the LG theory and methodology in order to obtain an accurate and complete language description;
- statistical techniques to record large scale data (i.e. corpus linguistics), and to verify phenomena in much shorter time than “manual” approaches;
- structured databases and knowledge bases, as ontologies, in order to process data not only from a strictly linguistic point of view, but also from logical and descriptive ones;

- NLP software mainly yet to build, considering that at the actual state of the art the only two packages existing are NooJ and Cataloga, (already discussed in section 2 and used to show LG textual analysis results in the experiments described further on).

In order to examine step-by-step individual elements of a potential and experimental NLP hybrid model, for each analytic approach we will here specify the features it should have and the specific tasks it should perform. First of all, we can delineate a scheme which summarizes the whole process. It starts from the analysis of linguistic data and arrives at different results, the most highly developed of which consist in the creation of structured lingware to be embedded in NLP applications, in order to provide linguistically enhanced solutions for KM platforms.

The scheme in Figure 7 shows an input consisting of a corpus of texts and unstructured linguistic data. These lexical items pass through an articulated and complex linguistic pre-processing phase in which, thanks to the theoretical elements of the proposed hybrid model (i.e. LG theory and methodologies of language formalization), linguistic data are processed and formalized according to three main tools: lexicon-grammar tables, electronic dictionaries and local grammars in form of automata (FSA/FST)[55]. So formalized and structured lingware becomes in turn the linguistic engine for NLP software. After this linguistic pre-processing phase we obtain many types of results which in part are put again in the loop and create a process, from time to time, more and more virtuous. The most immediate result concerns the parsing of the input text. Since some linguistic phenomena require more accurate and long-

[55] For more details on these three LG tools see paragraph 2.2.

term analyses, to avoid time consuming tasks and costly procedures in terms of human resources, at this stage and only to take advantage from short-term solutions, statistical techniques can be *a posteriori* applied to fill any procedural gap.

Outputs immediately usable are LR tagged in XML and bilingual lingware which can be integrated into applications of Computer Aided Translation (CAT), all of which allow the creation of NLP Apps to be deployed on KMS. Moreover Information Extraction (IE) and Text Classification (TC) activities can in turn enrich the information contained in the KMS, or better they can be used in order to populate ontologies. Therefore, ontological relations become a concrete part of the already mentioned virtuous loop, and also new matter to formalize by means of linguistic relationships (i.e. logical and semantic roles) and lexicon-grammar FST/FSA (see Figure 7).

An NLP Hybrid Model so described allows us to carry forward our experimental research focusing on a complex yet little explored problem, i.e. the one concerning MWUs treatment[56]. Considering that the analysis of large corpora highlights the massive presence of these linguistic forms, MWU recognition is to be considered as a crucial task for NLP activities.

To achieve this goal, we built a sample corpus, and in it we annotated all MWUs using an XML tagging: by means of NooJ[57], each compound word has been automatically tagged with the specific domain attributes of the field of knowledge of Medicine, in order to give semantic values

[56] A detailed classification of MWU types based on LG has just been discussed at the end of paragraph 1.

[57] For more specification on NooJ see the Section 2.

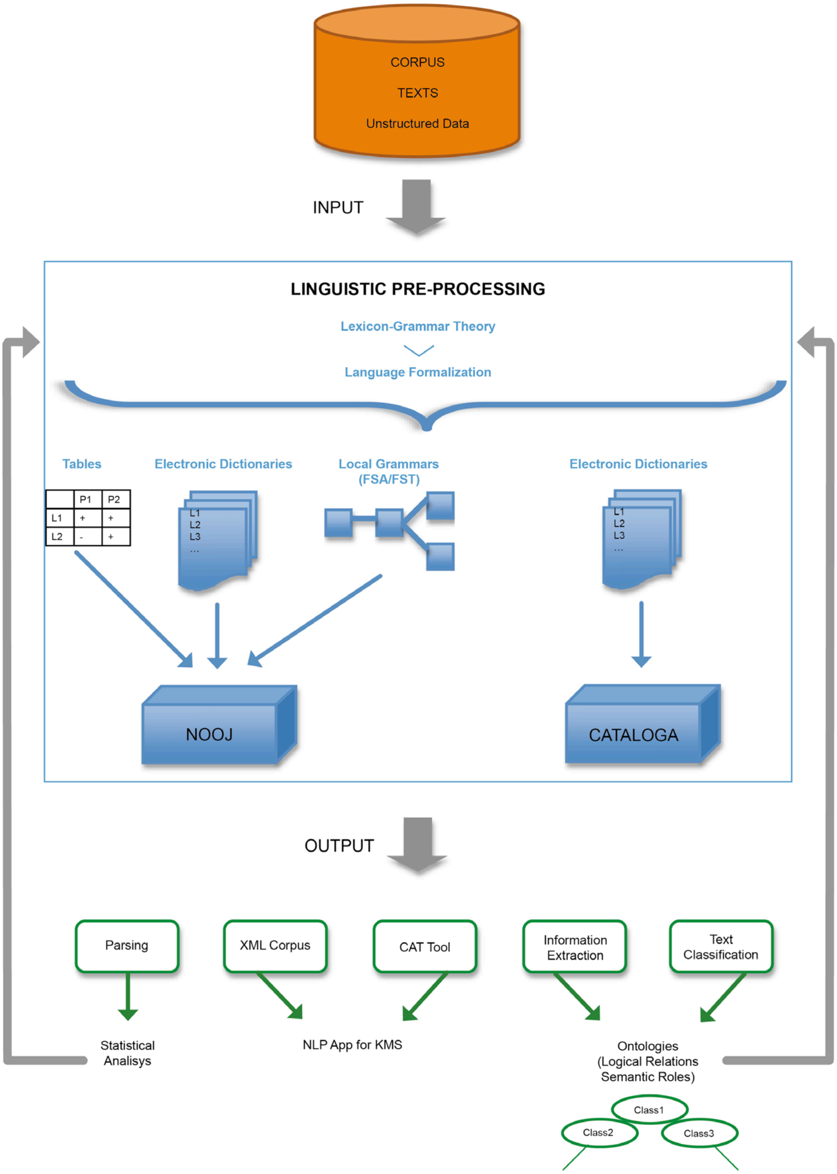


FIGURE 7. Hybrid Model for KMS based on NLP.

to tags[58]. Choosing a specific knowledge domain depends on the empirical observations in real-world texts, which highlight a strict necessity relation between MWUs and Terminology. It is possible to state that from a formal and semantic point of view, terminology fully exploits the procedures of compound word formation, in which a lexical element – for instance a noun with a generic meaning such as *vessel* – can be specified by adding other lexical elements, as happens with *lymph vessel, blood vessel, arterial vessel, venous vessel*, and so on. MWU recognition is also crucial in TC; to achieve it, we used Cataloga[59], a text classification software.

The linguistic model exploits an efficient and complete methodology for MWUs handling, which accounts for the description of the different types of MWUs and their semantic properties by means of well-defined steps: identification, interpretation, disambiguation and finally application. According to LG methods, this manually-based methodology develops accurate LRs useful to semiautomatic or automatic extraction and processing of MWUs in IR and NLP systems.

Nowadays, most computational linguistics approaches deal with MWUs treatment with reference to identification, formalization, disambiguation and application problems. Applying statistical rules in frequentist or probabilistic methods may collapse on MWUs issues, when processing not high-frequency ones in texts. In other cases, statistically-based parsers may not appropriately recognize strings of words as single meaning units, even if they are high-frequent; consequently, pieces of information are missed. Besides, many compound words (i.e.

[58] See Tim Berners-Lee, *Using labels to give semantics to tags* (2006-11-23) http://www.w3.org/DesignIssues/TagLabel.html.

[59] For more specification on Cataloga see the Section 2.

MWUs) are not recorded as lemmas in General Dictionaries unless they are high-frequent compounds, even if in real-word texts lexicon is mainly composed of MWUs (above all in specialized lexica). Furthermore, compounds change continuously both in number and in internal structure. This is the way we manually build continuously updated and well-crafted LR, in form of electronic dictionaries and local grammars, which are linguistically motivated and allows us to obtain accurate results for NLP purposes.

The assumptions underlying the creation of hybrid model provides a crucial first step that could be called "linguistic pre-processing of data" which exploits LG formal methods and linguistic software before developing any NLP application enriched also with ontologies and statistics' techniques. This crucial step will be shown in the following paragraph.

5.1 *Linguistic Pre-processing of Data for NLP Applications*

In the step of linguistic pre-processing of data we developed a bilingual (Italian/English) monitor corpus, formerly a part of the Medicine Manual edited by Merck Sharp & Dohme, available on line at http://www.msd-italia.it/altre/manuale/index.html. The current size of corpus is: 899048 word forms and 36370 different tokens[60].

By means of NooJ, MWUs were located inside the corpus and transformed into XML tags; each MWU was also automatically marked with

[60] The term "word forms" is here used with reference to any word recognized as belonging to a given language (in this case, Italian), be it canonical or inflected. On the contrary, the term "different tokens" refers to words (either simple and/or compound) counted only once during NLP analyses.

the label MED (i.e. the tag use for Medicine semantic domain). The analysis retrieved a 16% of MWUs (5,858 occurrences) on the total of different tokens, 66% (3,913 occurrences) of which are specific in the Medicine domain. Table 2 displays MWU POS patterns based on their morph-syntactic structure.

POS Pattern	# MWUs	% on the total (5,858)	# MED MWUs	% on the total of MED (3,913)
NA	4,089	69.80	2962	75.70
NPN	1,425	24.33	818	20.90
NN	157	2.68	108	2.76
AN	153	2.61	25	0.64
Others (Avv., Prep., etc.)	34	0.58	/	/

TABLE 2. Number of occurrences of MWUs by subcategory.

As previously mentioned[61], NooJ is a complex NLP environment in which it is possible to automatically read digitized texts, locating inside them specific linguistic patterns in the form of concordances. In the task just shown, in order to tag all MWUs, NooJ matched the corpus with the compound word electronic dictionary of Medicine, which contains almost 46,000 entries.

Basically, the tagging procedure is made possible by the specific structure of all terminological compound word electronic dictionaries

[61] See Section 2.

embedded in NooJ. The development and management of these electronic dictionaries consist of three main steps:

1. *Lexical acquisition*. During this on-going phase, MWUs are extracted from corpora and/or certified glossaries.
2. *Morpho-grammatical and syntactic tagging*. Each lexical entry is given an inflectional paradigm, in order to be inflected. The following string gives a sample of this morpho-grammatical formalization procedure:

   facce anteriori dell'iride, faccia anteriore dell'iride, N + Genere = f + Numero = p + Class = NAPN + Term = MED + Eng = facies anterior iridis, Class = NAN

   The tag "N" (noun) indicates the grammatical function of the whole compound. Other elements indicate the morpho-grammatical patterns of each compound structure, i.e.:

   - "NAPN" (noun + adjective + preposition + noun) for the internal structure;
   - "f" and "p" (feminine plural) give inflection indications;
   - "MED" (terminological tag) refers to Medicine knowledge domain.

3. *Testing on corpora*. The dictionary is used to automatically analyze and process large corpora.

In order to acquire information on compound words formation processes, we identify in the dictionary the typologies of MWU structure, as shown in the following table:

N° of constituents in the lexical unit	POS tags	Example
bi-gram	NA NN …	aborto spontaneo (MED) interfaccia utente (INF) …
tri-gram	NPN NPN NPN …	capacità del disco (INF) cassa di risparmio (ECON) morbo di Crohn (MED) …
fourth-gram	NAPN …	disturbo respiratorio del sonno (MED) …
fifth-gram	NPNPN …	disturbo da deficit di attenzione (MED) …
…	…	…

TABLE 3. Morpho-syntactic subcategories of MWUs.

The following excerpt taken from the Italian Electronic Dictionary of Medicine gives a sample of electronic dictionary string structure:

agenti patogeni, agente patogeno, N + Genere = m + Numero = p + Class = NA + Term = MED

flora residente, N + Genere = f + Numero = s + Class = NA + Term = MED

malattie infettive, malattia infettiva, N + Genere = f + Numero = p + Class = NA + Term = MED

pronto soccorso, N + Genere = m + Numero = s + Class = AN + Term = MED

quarto ventricolo, N + Genere = m + Numero = s + Class = AN + Term = MED

The LRs here described also consist of bilingual dictionaries useful in other NLP applications such as CAT and MT Systems. The following example represents a bilingual string extracted from the Italian-English dictionary of Medicine:

ubriachezze patologiche, ubriachezza patologica, N + Genere = f + Numero = p + Class = NA + Term = MED + Eng = pathologic intoxication, pathologic intoxication, Number = s+ Class = AN

uditi cromatici, udito cromatico, N + Genere = m + Numero = p + Class = NA+ Term= MED + Eng = chromatic audition, chromatic audition, Number = s+ Class = AN

uditi residui, udito residuo, N + Genere = m + Numero = p+ Class = NA + Term = MED + Eng = residual hearing, residual hearing, Number = s + Class = AN

When applied by means of NooJ to a bilingual corpus as the one which follows:

> […] I meccanismi di difesa includono le barriere naturali (p. es., la cute e le mucose) le risposte immuni aspecifiche (p. es., cellule fagocitarie [neutrofili, macrofagi] e i loro prodotti); e le risposte immuni specifiche (p. es., anticorpi). […]

these dictionaries can be used to automatically insert XML tags inside analysis outputs, as we can see in the text below, in which medicine MWUs have been tagged with their relevant morpho-grammatical labels:

> **I** <LU LEMMA=”meccanismo di difesa” CAT=”N” FLX=”C7” Genere=”m” Numero=”p” Class=”NPN” Term=”MED”>**meccanismi di difesa**</LU> **includono le barriere naturali (p. es., la cute e le mucose) le** <LU LEMMA=”risposta immune” CAT=”N” FLX=”C544” Genere=”f” Numero=”p” Class=”NA” Term=”MED”>**risposte immuni**</LU> **aspecifiche (p. es., cellule fagocitarie [neutrofili, mac-**

> **rofagi] e i loro prodotti); e le** <LU LEMMA=”risposta immune” CAT=”N” FLX=”C544” Genere=”f” Numero=”p” Class=”NA” Term=”MED”>**risposte immuni**</LU> **specifiche (p. es., anticorpi).**

As a sample, we explain the formalism of the XML labels used to tag the lemma “meccanismi di difesa”:

`CAT=”N”` >> Name (Part Of Speech)

`FLX=”C7”` >> morphologic automata used in NooJ to inflect the compound

`Genere=”m”` >> gender of the compound (masculine)

`Numero=”p”` >> number of the compuond (plural)

`Class=”NPN”` >> internal structure of the compound (Name + Preposition + Name)

`Term=”MED”` >> pertaining terminological/semantic domain of use (Medicine)

A further text classification task was performed on the above mentioned monitor corpus to highlight the relationship existing between domain terminology and MWUs, which are massively present in terminological texts. This task was achieved by means of Cataloga, which also works as a text classifier, i.e. matching terminological compound word electronic dictionaries and a given text, it achieves a classification on the basis of the semantic field(s) coped with.

Cataloga is a data mining software that can read text files and establish, without any human intervention:

- if a given text deals with a generic or a terminological topic;

- which is the eventual main specific knowledge domain dealt with in that text;
- as for the same text, if other terminological knowledge domains are dealt with, and which statistical relevance they have with reference to the main one.

This analytical inferential procedure is achieved by means of two specific elements:

- a lingware composed of terminological electronic dictionaries, both monolingual (Italian) and bilingual bidirectional (Italian-English); the entries of such dictionaries are all marked with terminological tags;
- two algorithms which achieves the matching between the text to analyze and the already mentioned electronic dictionaries.

All terminological words can be simultaneously recognized in one pass. Up to today, Cataloga has been used to analyze large and heterogeneous text corpora. The results achieved reach a 71% of correct analyses, a 29% of partially-correct analyses, and a 0% of incorrect analyses. Partially-correct analyses depend on electronic dictionaries that need continuous update. It is important to stress that Cataloga achieves detailed and successful analyses also with very short text files.

As already stated, Cataloga was applied to analyse our monitor corpus. The results obtained, and which are given in the following table, confirm the fact that it exists a strong link between terminology and the use/occurrence of MWUs:

Knowledge domain	MWUs (average %)
Medicine	76.47
Economics	4.99
Informatics	3.02
Law	2.51
Physics	1.09
Geography	0.65
Navigation	0.46
Zoology	0.28
Sciences & Techniques	0.25
Chemical	0.14
Hydrology	0.13
Optics	0.10
Microbiology	0.07
Other domains (Engineering, Astronomy, Psychology, Ecology, etc.)	0.02

TABLE 4. Average of MWUs classified in any knowledge domain.

In brief, the previous table shows that by recognizing the elevate presence of Medicine MWUs inside our corpus, Cataloga allows us to infer that it basically deals with topics associated to Medicine knowledge domain. Also, we can observe the presence in our corpus of MWUs belonging to other knowledge domains; Cataloga properly recognizes and duly classifies them thanks to the input electronic dictionary which includes about 180 semantic domains, also counting the one of Medicine[62].

5.1.1 *Linguistic Resources and Tools in Translation Processes*

In this paragraph we will briefly show how the LRs (such as domain electronic dictionaries) embedded in software as NooJ and Cataloga can be supporting tools for scientific or technical translation. The initial phases of this process imply several tasks that have to be performed by translators i.e., reading of the source text, identification of the main concepts and relevant terminology, documentary search using traditional documentary tools (paper dictionaries, thesauri, etc.) or web pages on the Internet, use of general, and specialized, monolingual, bilingual and multilingual electronic dictionaries on the Internet or on CD-ROM, consulting reference material provided by the customer or text corpora on the Internet or on CD-ROM, looking up information in a personal text corpus by means of text analysis or concordance software programs and updating and tailoring the linguistic resources or the translation tools according to the specific translation task that has to be performed.

[62] See paragraph 2.3 for more on the semantic fields used in Cataloga.

No tools are available on the market that speed up these complex and time-consuming activities.

The approach we would like to propose here is to introduce a higher degree of automation and integration for this crucial phase of the translation cycle which could also be beneficial to the subsequent translation phase.

An ideal documentary tool, in this respect, should contain a TM and IE facility from corpora which enables:

- document classification (identification of domain and extraction of relevant concepts) and automatic indexing based on linguistic information;
- retrieval of useful reference material by users such as appropriate terminology resources, parallel corpora, etc. which are automatically assigned to a specific translation project;
- pre-translation of the source text and/or updating of the translation tools (both MT and TM) with the relevant information found during the query phase.

This tool would allow users to semi-automate the translation analysis phase with regard to the retrieval of reference material (documents, terminology, corpora) for a particular translation project. Unlike state-of-the art collaborative translation workspaces, this would provide an advanced and indispensable feature based on linguistic knowledge within a typical translation workflow.

NooJ, indeed, could be a useful tool also in translation activities: it could use a bilingual electronic dictionaries and parallel corpora to tag texts (as shown above) that could also be employed for training purposes in conjunction with CAT, in specific MT system and TM applications, in order to identify and pre-translate linguistically significant phrases/clauses, with the aim of improving the computer-assisted translation results.

The following example is an extract of a translated corpus tagged in XML applying the procedure explained above:

> […] The defense mechanisms include natural barriers (eg., Skin and mucous membranes) non-specific immune responses (eg., Phagocytic cells [neutrophils, macrophages] and their products), and specific (p. es., antibodies). […]

> **The** <LU LEMMA=”defense mechanism” CAT=”N” Number=”p” Class=”NPN” Term=”MED”> **defense mechanisms**</LU> **include natural barriers (eg., Skin and mucous membranes) non-specific** <LU LEMMA=”immune response” CAT=”N” Number=”p” Class=”NA” Term=”MED”>**immune responses**</LU> **(eg., Phagocytic cells [neutrophils, macrophages] and their products), and specific** <LU LEMMA=”immune response” CAT=”N Number=”p” Class=”NA” Term=”MED”>**immune responses**</LU> **(p. es., antibodies)**.

In the CAT area, thanks to local grammars construction in form of transducers, it is possible to make MT of particular linguistic phenomena even when they are not provided in either electronic dictionary nor in bilingual parallel corpora. Figure 8 is a simplified graph that shows how the formal structure of an automaton. Nodes dedicated to domain-specific lexical items include bilingual domain terminology; this means that these nodes are connected to the terminological electronic dictionary. In the other nodes, however, more general grammatical labels are inserted which can co-occur in those specific contexts. Also, Figure 8 offers a simplified structure on which to base the construction of a FAQ automated responder for Medicine topics, integrated into a Web portal and allowing natural language queries:

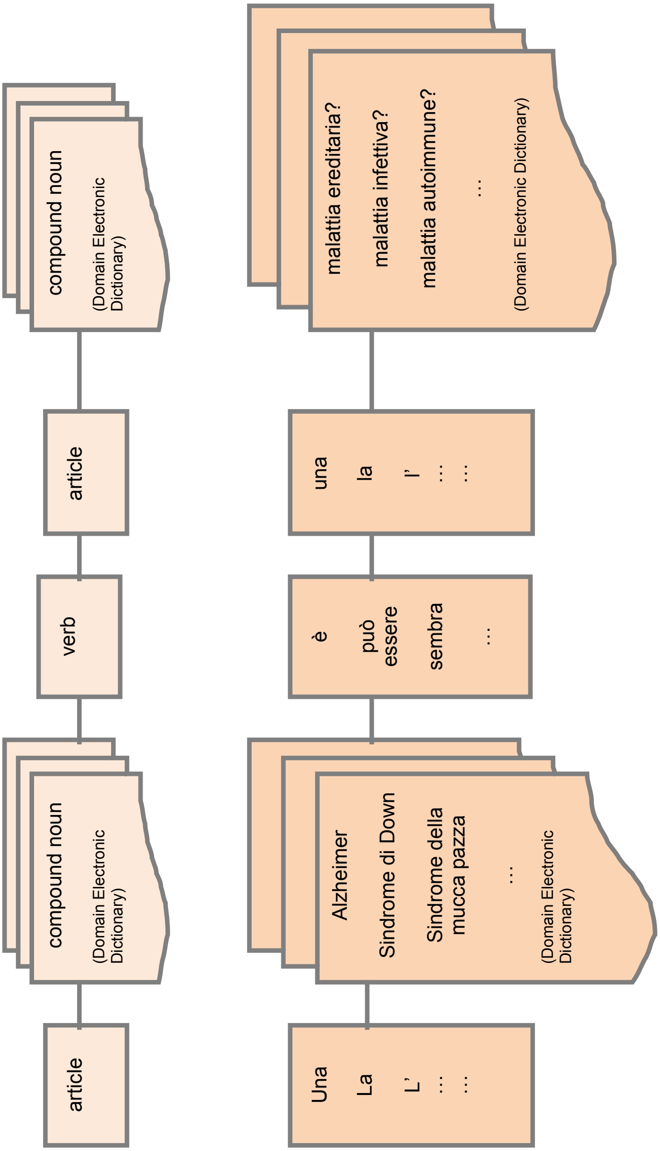


FIGURE 8. The graph shows an example of F.A.Q., a simplified structure for an Italian interrogative sentence, which includes domain terminology.

On the contrary, Figure 9 shows a more complex structure that models a graph for CAT activities; in it, nodes are linked to bilingual dictionaries, and are used to apply transformation routines to the structure, so allowing sentence translation from one language to another.

Also, we present below an experiment achieved with a bilingual version of Cataloga[63], which presents three additional features with reference to the monolingual version of the software, that is:

1. the listing of all the terminological occurrences, in decreasing order, classed on the basis of the relevant knowledge domains, together with their translation;
2. the tagging of all the terminological compound words, with their translation, in the source text, in XML format;
3. the automatic replacement of the translations in the target text.

The advantages coming from these may be summarized as follows:

1. the list of words obtained at the end of the text analysis process can be used in a specific crawling tool, such as BootCat[64] for

[63] This experiment of MT was presented at ASLIB 2011 Conference http://www.aslib.co.uk/conferences/tc_2011/ by the researcher Johanna Monti in a paper titled *In search of knowledge: text mining dedicated to technical translation*. (Monti, J., Elia A., Postiglione A., Monteleone M., Marano F., 2012 printing).

[64] Bootcat (http://bootcat.sslmit.unibo.it/?section=home) is an open source crawling tool that creates random tuples from a seed term list and runs a query for each tuple (on the Bing search engine). It constructs a URL list on the basis of the first 10 results obtained from the query and downloads the corresponding web pages. Bootcat is also available on the Sketchengine webpage (http://www.sketchengine.co.uk/?page=Website/SketchEngine) and as BootCat front-end, a web service front-end and a graphical user interface to the core tool, respectively.

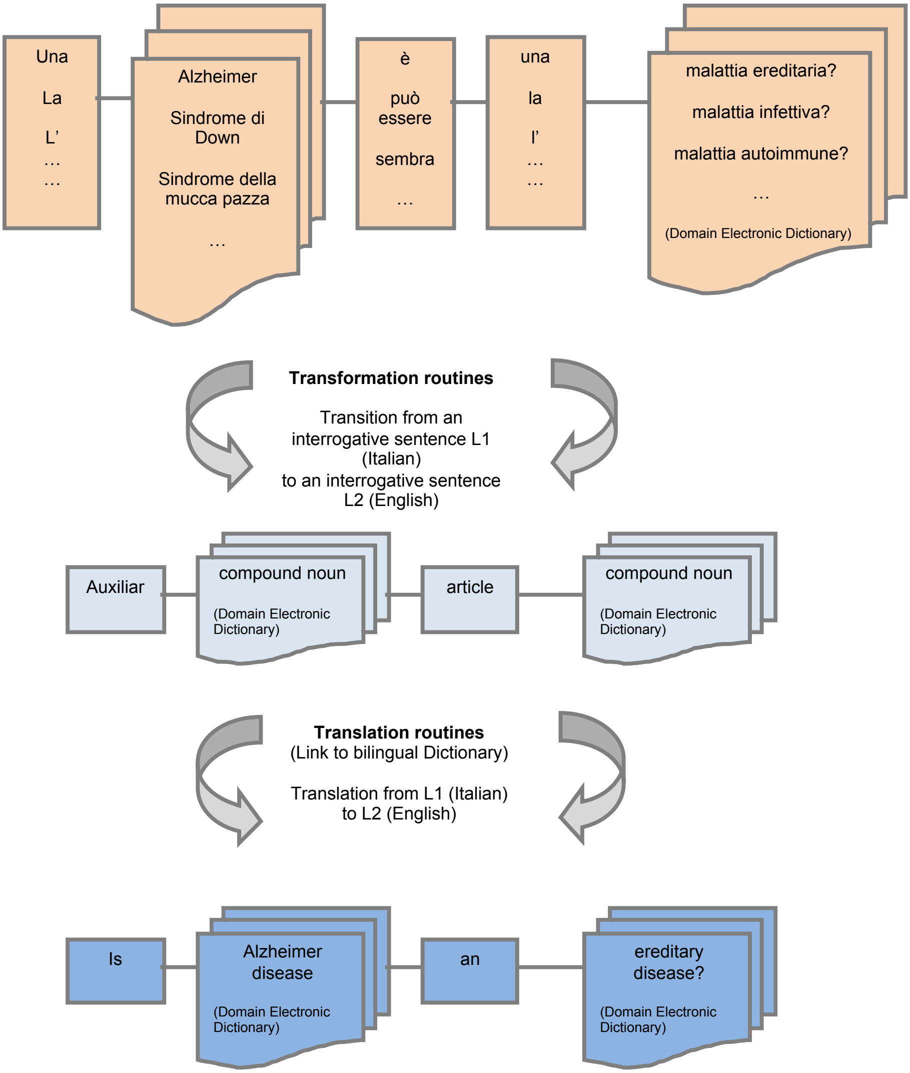

Una
La
L'
…
…
Alzheimer
Sindrome di Down
Sindrome della mucca pazza
…
è
può essere
sembra
…
una
la
l'
…
…
malattia ereditaria?
malattia infettiva?
malattia autoimmune?
…
(Domain Electronic Dictionary)
Transformation routines
Transition from an interrogative sentence L1 (Italian) to an interrogative sentence L2 (English)
Auxiliar
compound noun
(Domain Electronic Dictionary)
article
compound noun
(Domain Electronic Dictionary)
Translation routines
(Link to bilingual Dictionary)
Translation from L1 (Italian) to L2 (English)
Is
Alzheimer disease
(Domain Electronic Dictionary)
an
ereditary disease?
(Domain Electronic Dictionary)

FIGURE 9. The graphs show an example of CAT. Starting from an interrogative Italian sentence the tool transforms the structure into an interrogative English sentence and then translate it using domain terminology.

instance, to automatically retrieve useful reference material such as parallel or comparable corpora;

2. the tagged text can be used for training purposes in conjunction with MT, in specific SMT and TM applications, in order to identify and pre-translate linguistically significant phrases, with the aim of improving the computer-assisted translation results;
3. the pre-translated target text can be used as a basis during a traditional human-based translation cycle.

In order to provide a concrete example of how bilingual Cataloga processes texts and automatically extracts meanings, we will consider the following short passage which a human reader with an average cultural level could straightforwardly define as dealing with the field of Medicine:

> […] La vitamina A (Retinolo) svolge un'azione protettiva delle mucose e degli epiteti. Inoltre ha un ruolo nella crescita, favorendo lo sviluppo scheletrico. La carenza di vitamina A è una delle più comuni carenze vitaminiche. È comune soprattutto nei Paesi in via di sviluppo, rappresentando una della principali cause di cecità. La carenza di vitamina A è spesso dovuta a malassorbimento lipidico, ad alcolismo, e si osserva più comunemente negli anziani. Un sintomo precoce di carenza di vitamina A è la cecità notturna, seguita da secchezza della congiuntiva, macchie di Bitot (macchie biancastre della sclera). Questa risposta fatta da me su altro sito le fa capire a che cosa è dovuta la macchia di Bitot e di che colore è ovvero biancastro. La sua sembra più o un piccolo nevo nevocellulare piano oppure una zona di assottigliamento sclerale, completamente innocua e sine materia dal punto di vista patologico, che lascia intravedere

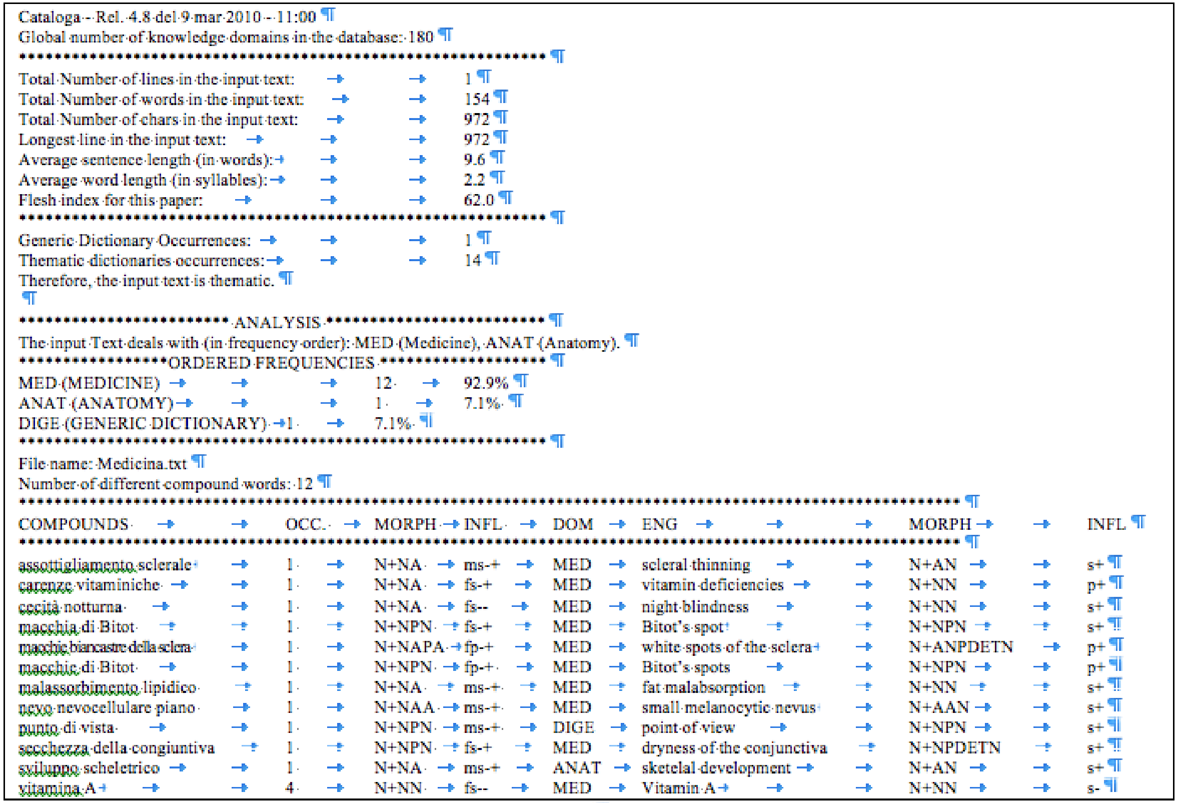
```
    la componente bluastra sottostante. […] [65]
```

After reading and analysing it, bilingual Cataloga automatically produces a table with the results of the text processing (see Table 5).

This table shows that analyzing and computing the terminological compound words occurring in it Cataloga has inferred that the input text deals with Medicine, i.e. it has reached the same conclusions as the human reader (similar results were also obtained in the experiment of paragraph 5.1).

The bilingual list of terminological compound words obtained by Cataloga can be used to automatically produce a precise and specific list of "seed terms", both in the source and in the target language, tailored on the source text, to be used in queries on the Web with a crawling tool. For our experiment we used the BootCat toolkit (Baroni *et al.*, 2004), a well-known suite of Perl scripts for bootstrapping specialized language corpora from the web.

[65] […] Vitamin A (Retinol) exerts a protective action on the mucous membranes and epithets. It also has a role in growth, supporting skeletal development. Lack of vitamin A is one of the most common vitamin deficiencies. It is especially common in developing countries, representing one of the main causes of blindness. Vitamin A deficiency is usually due to fat malabsorption and alcoholism and is most commonly seen in elderly people. An early sign of vitamin A deficiency is night blindness, followed by dryness of the conjunctiva and Bitot's spots (white spots on the sclera). This answer I gave on another site helps you understand the origins of Bitot's spots, and their colour, i.e. whitish. Your spot looks more like a small melanocytic nevus or a scleral thinning area that is completely harmless and sine materia from a pathological point of view, with a bluish part underneath. […] (Available on http://www.medicitalia.it/consulti/Oculistica/65819/Macchianella-sclera; English translation by the author).



```
Cataloga - Rel. 4.8 del 9 mar 2010 - 11:00
Global number of knowledge domains in the database: 180
*************************************************************
Total Number of lines in the input text:                1
Total Number of words in the input text:                154
Total Number of chars in the input text:                972
Longest line in the input text:                         972
Average sentence length (in words):                     9.6
Average word length (in syllables):                     2.2
Flesh index for this paper:                             62.0
*************************************************************
Generic Dictionary Occurrences:                         1
Thematic dictionaries occurrences:                      14
Therefore, the input text is thematic.

********************** ANALYSIS **********************
The input Text deals with (in frequency order): MED (Medicine), ANAT (Anatomy).
****************ORDERED FREQUENCIES *****************
MED (MEDICINE)                          12       92.9%
ANAT (ANATOMY)                          1        7.1%
DIGE (GENERIC DICTIONARY)  1            7.1%
*************************************************************
File name: Medicina.txt
Number of different compound words: 12
*****************************************************************************************
COMPOUNDS                  OCC.   MORPH    INFL   DOM    ENG                          MORPH         INFL
*****************************************************************************************
assottigliamento sclerale  1      N+NA     ms-+   MED    scleral thinning             N+AN          s+
carenze vitaminiche        1      N+NA     fs-+   MED    vitamin deficiencies         N+NN          p+
cecità notturna            1      N+NA     fs--   MED    night blindness              N+NN          s+
macchia di Bitot           1      N+NPN    fs-+   MED    Bitot's spot                 N+NPN         s+
macchie biancastre della sclera 1 N+NAPA   fp-+   MED    white spots of the sclera    N+ANPDETN     p+
macchie di Bitot           1      N+NPN    fp-+   MED    Bitot's spots                N+NPN         p+
malassorbimento lipidico   1      N+NA     ms-+   MED    fat malabsorption            N+NN          s+
nevo nevocellulare piano   1      N+NAA    ms-+   MED    small melanocytic nevus      N+AAN         s+
punto di vista             1      N+NPN    ms-+   DIGE   point of view                N+NPN         s+
secchezza della congiuntiva 1     N+NPN    fs-+   MED    dryness of the conjunctiva   N+NPDETN      s+
sviluppo scheletrico       1      N+NA     ms-+   ANAT   sketelal development         N+AN          s+
vitamina A                 4      N+NN     fs--   MED    Vitamin A                    N+NN          s-
```

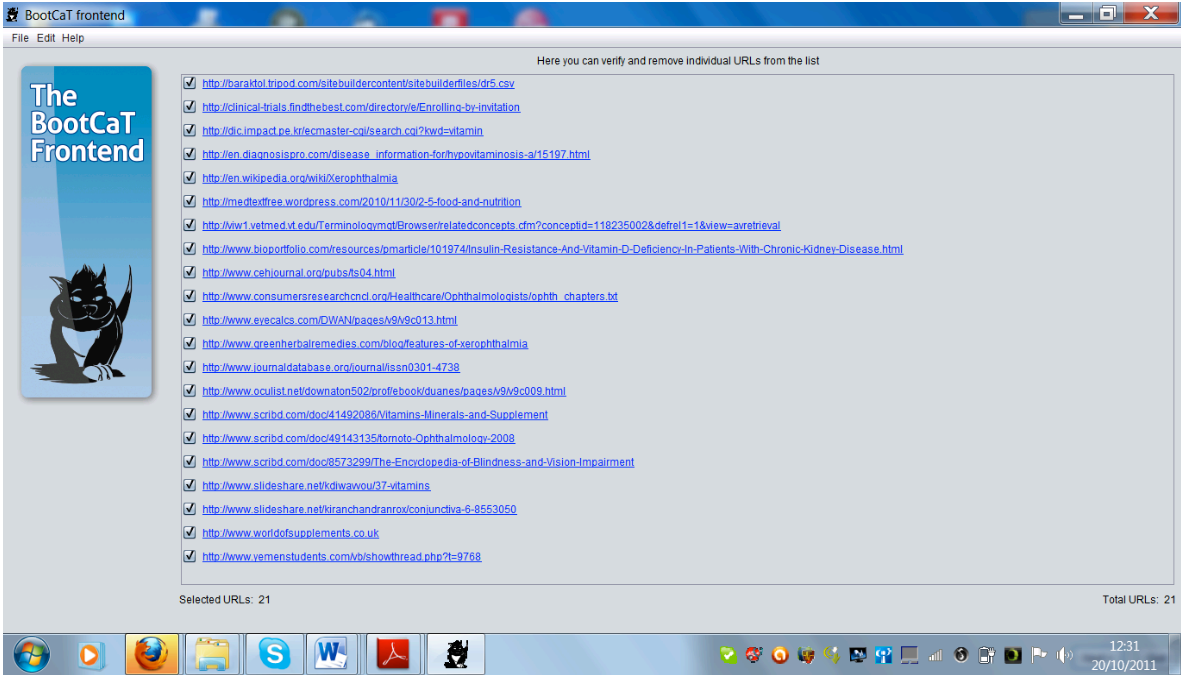
TABLE 5. Bilingual Cataloga Analysis Results.

Taking as input the key terms extracted by means of the automatic text analysis procedure performed by Cataloga, BootCat draws upon web data to automatically build a specialised corpus for the domain of interest and tailored on the specific text to be translated. In this way, the most relevant web pages which specifically refer to the subject matter of the text to be translated can be collected.

For instance, if we take the list of English terminological compound words (refer to Table 5) produced during the text analysis phase illustrated in the previous section and we use it as ‘seed terms’ in Bootcat, we obtain the list of web sites as indicated in Figure 10.

The list of Web sites contains relevant information sources such as medical texts, glossaries, thesauri and text corpora related to the subject matter of the analysed text.

In addition to the above mentioned options for integrating Cataloga in a translation environment, a further possibility, already available, is to use the list of compound words generated during the analysis of the source text performed by Cataloga to pre-translate the source text, thereby ensuring a coherent use of terminology throughout the whole target text (see Figure 11).



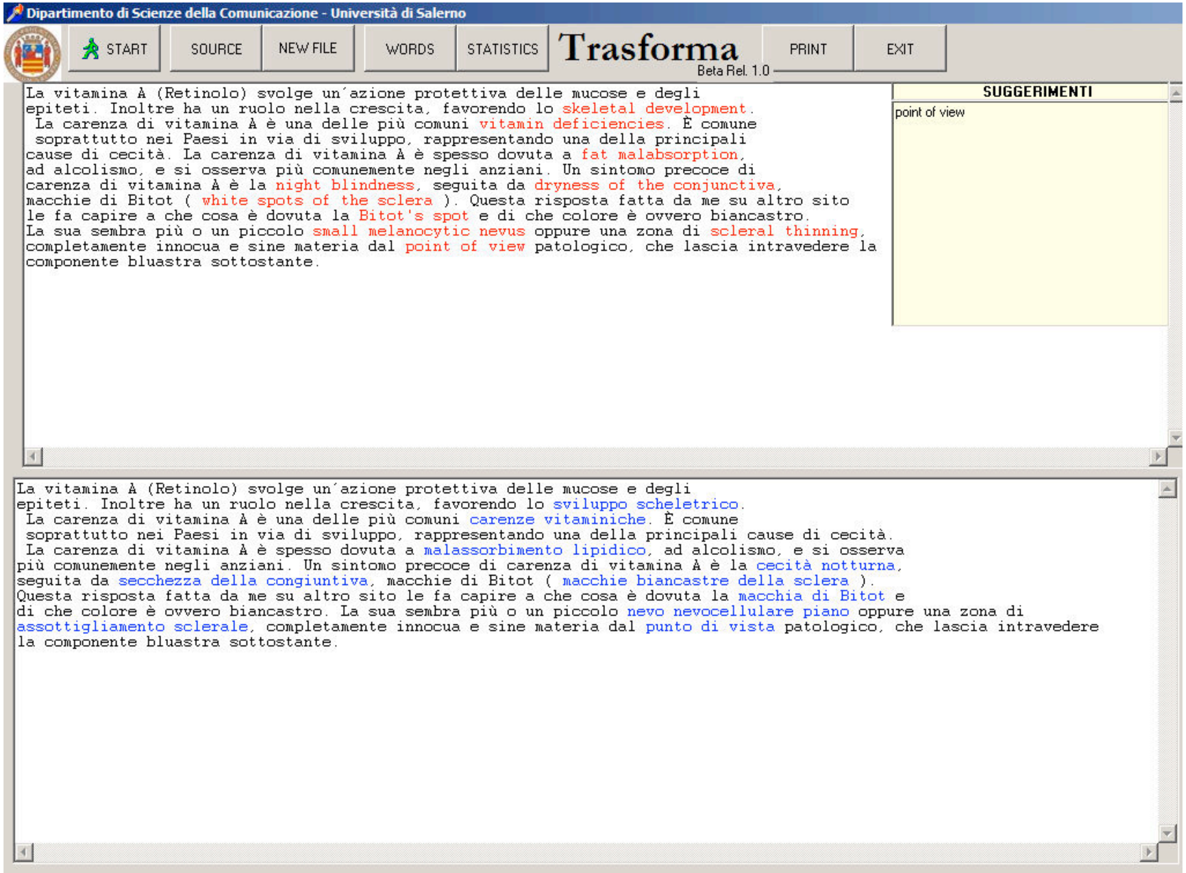
Here you can verify and remove individual URLs from the list

- ☑ http://baraktol.tripod.com/sitebuildercontent/sitebuilderfiles/dr5.csv
- ☑ http://clinical-trials.findthebest.com/directory/e/Enrolling-by-invitation
- ☑ http://dic.impact.pe.kr/ecmaster-cgi/search.cgi?kwd=vitamin
- ☑ http://en.diagnosispro.com/disease_information-for/hypovitaminosis-a/15197.html
- ☑ http://en.wikipedia.org/wiki/Xerophthalmia
- ☑ http://medtextfree.wordpress.com/2010/11/30/2-5-food-and-nutrition
- ☑ http://viw1.vetmed.vt.edu/Terminologymgt/Browser/relatedconcepts.cfm?conceptid=118235002&defrel1=1&view=avretrieval
- ☑ http://www.bioportfolio.com/resources/pmarticle/101974/Insulin-Resistance-And-Vitamin-D-Deficiency-In-Patients-With-Chronic-Kidney-Disease.html
- ☑ http://www.cehjournal.org/pubs/ts04.html
- ☑ http://www.consumersresearchcncl.org/Healthcare/Ophthalmologists/ophth_chapters.txt
- ☑ http://www.eyecalcs.com/DWAN/pages/v9/v9c013.html
- ☑ http://www.greenherbalremedies.com/blog/features-of-xerophthalmia
- ☑ http://www.journaldatabase.org/journal/issn0301-4738
- ☑ http://www.oculist.net/downaton502/prof/ebook/duanes/pages/v9/v9c009.html
- ☑ http://www.scribd.com/doc/41492086/Vitamins-Minerals-and-Supplement
- ☑ http://www.scribd.com/doc/49143135/tornoto-Ophthalmology-2008
- ☑ http://www.scribd.com/doc/8573299/The-Encyclopedia-of-Blindness-and-Vision-Impairment
- ☑ http://www.slideshare.net/kdiwavvou/37-vitamins
- ☑ http://www.slideshare.net/kiranchandranrox/conjunctiva-6-8553050
- ☑ http://www.worldofsupplements.co.uk
- ☑ http://www.yemenstudents.com/vb/showthread.php?t=9768

Selected URLs: 21

Total URLs: 21

FIGURE 10. URL list generated on the basis of the Cataloga list of compound words.



```
La vitamina À (Retinolo) svolge un´azione protettiva delle mucose e degli
epiteti. Inoltre ha un ruolo nella crescita, favorendo lo skeletal development.
 La carenza di vitamina À è una delle più comuni vitamin deficiencies. È comune
 soprattutto nei Paesi in via di sviluppo, rappresentando una della principali
cause di cecità. La carenza di vitamina À è spesso dovuta a fat malabsorption,
ad alcolismo, e si osserva più comunemente negli anziani. Un sintomo precoce di
carenza di vitamina À è la night blindness, seguita da dryness of the conjunctiva,
macchie di Bitot ( white spots of the sclera ). Questa risposta fatta da me su altro sito
le fa capire a che cosa è dovuta la Bitot's spot e di che colore è ovvero biancastro.
La sua sembra più o un piccolo small melanocytic nevus oppure una zona di scleral thinning,
completamente innocua e sine materia dal point of view patologico, che lascia intravedere la
componente bluastra sottostante.
```

point of view

```
La vitamina À (Retinolo) svolge un´azione protettiva delle mucose e degli
epiteti. Inoltre ha un ruolo nella crescita, favorendo lo sviluppo scheletrico.
 La carenza di vitamina À è una delle più comuni carenze vitaminiche. È comune
 soprattutto nei Paesi in via di sviluppo, rappresentando una della principali cause di cecità.
 La carenza di vitamina À è spesso dovuta a malassorbimento lipidico, ad alcolismo, e si osserva
più comunemente negli anziani. Un sintomo precoce di carenza di vitamina À è la cecità notturna,
seguita da secchezza della congiuntiva, macchie di Bitot ( macchie biancastre della sclera ).
Questa risposta fatta da me su altro sito le fa capire a che cosa è dovuta la macchia di Bitot e
di che colore è ovvero biancastro. La sua sembra più o un piccolo nevo nevocellulare piano oppure una zona di
assottigliamento sclerale, completamente innocua e sine materia dal punto di vista patologico, che lascia intravedere
la componente bluastra sottostante.
```

FIGURE 11. Pre-translation of the source text using the Cataloga bilingual compound word list.

Discussions and Conclusions

At the conclusion of this research, we feel that some further clarifications are necessarily to be made concenring the general purposes of the analyses conducted and the validity of the results obtained.

The first point is: we are conscious of the fact that natural language is not, in absolute formalizable in its entirety, and that due to its not completely systematic nature – as stated by Steven Pinker, human language is not the means best way to express one's thoughts – some areas of linguistic production are difficult to manage automatically, also and even in spite of the method of formalization adopted. However, in this sense, borrowing the famous Pareto's Law, we can credibly affirm that the joint adoption of the methods here outlined could lead to a correct formalization and automation of more than eighty percent of the "content" of a given language. Certainly, the treatment of the remaining part would be difficult to manage, but the detailing level provided by the NLP tools with which we have been dealing in these pages comforts us in our thinking that the future challenges posed by natural language and NLP will be tackled by suitable means. The second point is: the revolutionary ideas of Tim Berners-Lee on the Semantic Web and its structure have initially given rise to a form of euphoric and optimistic research, as often happens with all the epistemological innovative approaches; in the '50s, the same happened for example with the TGG of Noam Chomsky. It is known however that best scientific revolutions are long ones, or those that confirm their validity over the years, and in some cases, if we think of Galileo, Newton and Einstien, over the centuries.

So, considering the lack of attention that until a few years ago has been given by (great and small) Web content managers to natural language, it is not surprising that today the recent "dream" by Tim Berners-Lee of a global Semantic Web has already stalled, and has been diluted in different subsets of methodological analysis, such as the one of Linked Data. Obviously, we have neither the means nor the purpose of discrediting the original scientific visions on the Semantic Web, of which we are staunchest supporters; however, basing our assumptions on the reflection that every human activity is difficult to be widely shared unless it is not expressed via natural language, then we choose once again to affirm that only if we pay careful attention to natural language we will be albe to structure intelligent computerized systems capable of interacting with humans using natural language in a consistent and effective way. Our final reflection is also based on the example provided by current search engines: having since their inception underestimated the impact that natural language autonomous peculiarities might and may have on methods and tools for the automatic processing of linguistic data, today search engines often find themselves unable to drastically improve their procedures of query and IR, of which we all know the limits. We must point out that Google was born in 1998, when Lexicon-Grammar, to give just one example, had already existed for over thirty years. If the designers of Google had initially entrusted themselves to a well-structured natural language formalization method, perhaps the Semantic Web today would not be just a project, but a concrete reality.

However, retrospectively analyzing actual stalemates never brings good fruits or solutions. It is for this reason that in our future research path we hope to continue the line described in these pages, with the aim to explore the themes and instruments we have discussed, and having the honesty to acknowledge our methodological and procedural errors, when they will turn out before us.

References

Abney, S. (1997). *Part-of-speech tagging and partial parsing. Corpus-Based Methods in Language and Speech Processing*. Dordrecht, Holland: Kluwer Academic Publishers.

Allemang, D. (2006). *Ontologies, Reuse and Domain Analysis.* TopQuadrant, Inc.

Alshawi, H. (1994). Qualitative and quantitative models of speech translation. *Proceedings of the Workshop on Combining Symbolic and Statistical Approaches to Language*.

Baayen, Harald R. (2008). *Analyzing Linguistic Data: A Practical Introduction to Statistics Using R*. Cambridge University Press.

Baker, Collin F., Fillmore C.J., Cronin B. (2003). The Structure of the Framenet Database. *International Journal of Lexicography* 16.3, pp. 281-296.

Baroni, M. & Bernardini S. (2004). Boot-CaT: Bootstrapping corpora and terms from the web. *Proceedings of LREC 2004*, Lisbon: ELDA. (2004): 1313–1316.

Benzécri, Jean P. (1963). *Cours de linguistique mathématique*. Rennes: Université de Rennes.

Benzécri, Jean P. (1973). *L'Analyse des données* (2 tomes). Paris: Dunod.

Benzécri, Jean P. (1982). *Histoire et préhistoire de líanalyse des donne.* Paris: Dunod.

Benzécri, Jean P. *et al*. (1981). *Pratique de líanalyse des données - Linguistique et lexicologie*. Paris: Dunod.

Berners-Lee, Tim J., Hendler J., Lassila O. (2001). The Semantic Web. *Scientific American*, May, pp. 28-37.

Bird, S., Liberman M. (1999). A formal framework for linguistic annotation. *Technical Report MS-CIS-99-01*, Department of Computer and Information Science, University of Pennsylvania. [xxx.lanl.gov/abs/cs.CL/9903003], expanded from version presented at ICSLP-98, Sydney, revised version to appear in *Speech Communication*.

Bloomfield, L. (1933). *Language*. New York: Henry Holt.

Bolasco, S. (1999). *Analisi multidimensionale dei dati.* Roma: Carocci Editore.

Bolasco, S. (2005). Statistica testuale e text mining: alcuni paradigmi applicativi. *Quaderni di Statistica Vol. 7*.

Bolshakov, Igor A. & Gelbukh A. (2004). *Computational Linguistics. Models, Resources, Applications*. Mexico DF: Instituto Politécnico Nacional.

Bontcheva, K., Dimitrov M., Maynard D., Tablan V., Cunningham H. (2002). Shallow Methods for Named Entity Coreference Resolution. *Chaînes de références et résolveurs d'anaphores, workshop TALN 2002*, Nancy, France.

Bontcheva, K., Kiryakov A., Cunningham H., Popov B., Dimitrov M. (2003). Semantic Web Enabled, Open Source Language Technology. In *Proceedings of EACL Workshop "Language Technology and the Semantic Web", NLPXML-2003*, 13 April, 2003.

Botha, A., Kourie D., Snyman R. (2008). *Coping with Continuous Change in the Business Environment, Knowledge Management and Knowledge Management Technology*, Chandice Publishing Ltd.

Brickley, D., Guha, R. (2002). RDF Vocabulary Description Language 1.0: RDF Schema. Technical report, W3C Working Draft. http://www.w3.org/TR/rdfschema/.

Brown, J.S., Duguid P. (1991). Organizational Learning and Communities of Practice. Toward a Unified View of Working, *Organization Science* vol.2, no.1.

Brown, P., Cocke J., Della Pietra S., Della Pietra V., Jelinek F., Mercer R. and Roossin P. (1988). A statistical approach to language translation. In Dénes Vargha, (ed.) *Coling 88*: *Proceedings of the 12th conference on Computational linguistics, volume 1. Budapest: John Von Neumann society for computing sciences*. pp. 71–76.

Carletta, J., Kilgour J., O'Donnell T.J., Evert S., Voormann H. (2003). The NITE object model library for handling structured linguistic annotation on multimodal data sets. In *Proceedings of the EACL Workshop on Language Technology and the Semantic Web (NLPXML 2003)*. Budapest, Hungary, April 2003.

Carr, L., Bechhofer S., Goble C., Hall W. (2001). Conceptual Linking: Ontology-based Open Hypermedia. In *The WWW10 Conference*, Hong Kong, May, pp. 334-342.

Chang, S.K. (2010). A General Framework for Slow Intelligence Systems. *International Journal of Software Engineering and Knowledge Engineering* 20(1): 1-15.

Chomsky, Noam A. (1957). Syntactic Structures. Paris: Mouton, The Hague.

Chomsky, Noam A. (1965). Aspects of the Theory of Syntax. Cambridge, Massachusetts: MIT Press.

Chomsky, Noam A. (1968) *Language and Mind, New York, Harcourt Brace Jovanovich, Inc.*

Chomsky, Noam A. (1993). A minimalist program for linguistic theory. Hale, Kenneth L. and S. Jay Keyser, (eds.) *The view from Building 20: Essays in linguistics in honor of Sylvain Bromberger*. Cambridge, MA: MIT Press. 1-52.

Chomsky, Noam A. (1995). *The Minimalist Program*. Cambridge, Mass.: The MIT Press.

Choueka, Y. (1998). Looking for needles in a haystack or locating interesting collocational expressions in large textual database. *Proceedings of the RIAO*, pp. 38-43.

Cimiano, P., Haase P., Herold M., Mantel M. and Buitelaar P. (2007). LexOnto: A model for ontology lexicons for ontology-based NLP. *Proceedings of the ISWC'07 OntoLex Workshop*.

Colace, F., Chang S.K., De Santo M. (2010). SINMS: A Slow Intelligence Network Manager based on SNMP Protocol. *DMS* 2010: 47-52.

Collier, N., Takeuchi K., Kawazoe A. (2003). Open Ontology Forge: An Environment for Text Mining in a Semantic Web World. In *Proceedings of the International Workshop on Semantic Web Foundations and Application Technologies*, Nara, Japan, 11th March.

Cunningham, H. (1999). *Information Extraction: a User Guide* (revised version). Department of Computer Science, University of Sheffield, May.

Cunningham, H., Maynard D. and Tablan V. (2000). JAPE: a Java Annotation Patterns Engine (Second Edition). *Technical report CS--00--10*, Univ. of Sheffield, Department of Computer Science.

Cunningham, H., Maynard D., Bontcheva K. and Tablan V. (2002). GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*.

D'Agostino, E. (1992). *Analisi del discorso. Metodi descrittivi* *dell'italiano d'uso*. Napoli: Loffredo.

D'Agostino, E., Elia A. (1998). Il significato delle frasi: un continuum dalle frasi semplici alle forme polirematiche. In AA.VV, *Ai limiti del linguaggio*. Bari: Laterza, p. 287-310.

Dantzig, George B. (1940). *On the non-existence of tests of "Student's" hypothesis having power functions independent of σ*. Annals of Mathematical Statistics, Volume 11, numero 2, pp. 186-192.

Dantzig, George B. and Wald A. (1951). *On the Fundamental Lemma of Neyman and Pearson*. Annals of Mathematical Statistics, No. 22; pp. 87-93.

Dhillon, I.S. & Modha D.S. (2001). Concept decompositions for large sparse text data using clustering. *Machine Learning*, 42(1-2):143–175.

Dimitrov, M., Bontcheva K., Cunningham H., Maynard D. (2002). A Light-weight Approach to Coreference Resolution for Named Entities in Text. *Proceedings of the Fourth Discourse Anaphora and Anaphor Resolution Colloquium (DAARC)*, Lisbon, September 2002.

Downing, P. (1977). On the creation and use of English compound nouns. *Language* Vol. 53, pp. 810-842.

Drucker, Peter F. (1969). *The Age of Discontinuity, Guidelines to Our Changing Society*. New York: Harper & Row.

Elia, A. (1978). Pour un lexique-grammaire de la langue italienne: les complétives objet. *Linguisticae Investigationes*, II, 2, Amsterdam/Philadelphia: John Benjamins.

Elia, A. (1984). *Le verbe italien*. Bari/Paris: Schena-Nizet.

Elia, A., Langella A.M., Marano F., Monteleone M., Sabatino S., Vellutino D. (2010) *Manually constructed lexicons and grammars for NLP: building lingware for Smart Information Retrieval Systems.* In Vitas, Dusko and Krstev, Cvetana (eds.) Proceedings of The Lexis and Grammaire

Conference 2010, Belgrade Serbia 15-18 settembre 2010, Belgrade: Faculty of Mathematics, University of Belgrade, Studentski trg 16, pp. 131-140.

Elia, A., Marano F., Monteleone M., Sabatino S., Vellutino D. (2010). *Strutture lessicali delle informazioni comunitarie all'interno di domini specialistici*. In Bolasco, Chiari, Giuliano (eds.), *Statistical Analysis of Textual Data, Proceedings of 10th International Conferences "Journées D'Analyse Statistique des Données Textuelles"*. LA SAPIENZA - University of Rome Italy, 9 - 11 June 2010, MILANO: LED Ed. Universitarie Lettere Economia Diritto, vol. 2, p. 1227-1236.

Elia, A., Monteleone M, De Bueriis G. & Di Maio F. (2008). Le polirematiche dell'italiano. Elia, Annibale & Giustino De Bueriis (eds.). *Lessici elettronici e descrizioni semantiche, sintattiche e morfologiche: Progetto PRIN 2005 Atlanti Tematici Informatici - ALTI, Collana "Lessici & Combinatorie", n. 2, Dipartimento di Scienze della Comunicazione dell'Università degli Studi di Salerno*. Salerno: Plectica, p. 11-65.

Elia, A., Monteleone M., Marano F. (printing) Starting by the concept of transformation in Harris and Chomsky until lexique-grammaire of Maurice Gross. *Proceedings of ICHOLS XII – 2th International Conference on the History of the Language Sciences.* 29 August-1 September 2011, St Petersburg, Russia.

Elia, A., Postiglione A., De Bueriis G., Monteleone M., Marano F. (2010) *Semantics from Lexis Grammar* In Vitas, Dusko and Krstev, Cvetana (eds.) Proceedings of The Lexis and Grammaire Conference 2010, Belgrade Serbia 15-18 settembre 2010, Belgrade: Faculty of Mathematics, University of Belgrade, Studentski trg 16, pp. 121-130.

Elia, A., Postiglione A., Monteleone M. (2010). *Cataloga. Sistema informatico per la catalogazione automatica di testi*, Release 4.8., Software.

Elia, A., Vellutino D., Marano F., Langella A.M., Napoli A. (2011). Semantic Web and language resources for e-Government: linguistically motivated Data Mining. In Rajendra Akerkar (eds.) *Proceedings of International*

Conference on WIMS – Web Intelligence, Mining and Semantics, 2011, 25-27 May, Songdal Norway. Published by ICPS – ACM.

Elia, A., Vietri S. (2001). Analisi automatica dei testi e dizionari elettronici. In R. Cordeschi, E. Burattini, (eds.). *Intelligenza Artificiale. Manuale per le discipline della comunicazione*, Carocci Editore: Roma, pp. 203-226.

Elia, A., Vietri S., Postiglione A., Monteleone M., Marano F. (2010). Data Mining Modular Software System. In Arabnia H.R., Marsh A., Solo A.M.G *Proceedings of The 2010 International Conference on Semantic Web & Web Services, WorldComp 2010 Conference*, July 12-15 2010, Las Vegas Nevada USA, CSREA Press, pp. 127-133.

EMDA – Elia, A., Martinelli M., D'Agostino E. (1981). *Lessico e strutture sintattiche.* Napoli: Liguori.

Evert, S., Carletta J., O'Donnell T.J., Kilgour J., V ogele A., Voormann H. 2003. *The NITE Object Model*. Version 2.1, March 24, 2003. Available on line http://www.google.it/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0CDgQFjAB&url=http%3A%2F%2Fwww.ltg.ed.ac.uk%2FNITE%2Fdocuments%2FNiteObjectModel.v2.1.pdf&ei=wURwT5XDNPH74QSbnZC_Ag&usg=AFQjCNHJzPKTx0ystt3iNewIY-mqlCv0Pg.

Ferrari, G. (2004). A State of the art in Computational Linguistics. *Proceedings of the 17th International Congress of Linguists – Prague*. Amsterdam: John Benjamins.

Ferrari, G. (2005). La ricerca in Linguistica Computazionale tra modelli formali ed analisi empirica. In G.Marotta (ed.) *Atti del Convegno di Studi in memoria di Tristano Bolelli*, in *Studi e Saggi Linguistici*, XL-XLI (2002-2003), Pisa, pp.101-119.

Fonseca, C. (2010). Notes from the Field. The Digital Divide and the Cognitive Divide: Reflections on the Challenge of Human Development in the Digital Age Digital Revolution, Digital Divide. *Information Technologies*

& International Development Journal, Volume 6, SE, Special Edition, pp. 25-30.

Frank, U. (2002). A Multilayer Architecture for Knowledge Management Systems. In Barnes, S. (eds.), *Knowledge Management Systems: Theory and Practice.* Thomsen Learning.

Gamble, P.R., Blackwell J. (2001). *Knowledge Management: A State of the Art Guide*. Kogan Page Ltd.

Girju, R., Moldovan D., Tatu M., Antohe D. (2005). On the semantics of noun compounds. *Computer Speech and Language*, 19:479-496.

Graffi, G. (2001). *200 Years of Syntax*. Amsterdam/Philadelphia: John Benjamins.

Grishman, R. (1986). *Computational Linguistics: An Introduction*. Cambridge University Press.

Gross, M. (1963). Linguistique mathématique et langages de programmation. *Revue française de traitement de l'information*, 4, p.231-253.

Gross, M. (1964). Sur certains procédés de définition de langages formels. *Automata Theory*, New York: Academic Press, p.181-200.

Gross, M. (1968). L'emploi des modèles en linguistique. *Langages* 9, Paris: Larousse, p.3-8.

Gross, M. (1972). *Mathematical Models of Language*. Englewood Cliffs, N.J.: Prentice-Hall.

Gross, M. (1975). *Méthodes en syntaxe*, Paris: Hermann.

Gross, M. (1979). On the Failure of Generative Grammar. *Language*, Vol. 55, No. 4, pp. 859-885. Linguistic Society of America.

Gross, M. (1981). Les bases empiriques de la notion de prédicat sémantique. In A. Guillet and C. Leclère, (eds.), *Formes Syntaxiques et Prédicats Sémantiques*. *Langages* volume 63, pages 7-52. Larousse, Paris.

Gross, M. (1984). Lexicon-grammar and the syntactic analysis of French. *ACL '84 Proceedings of the 10th International Conference on Computational Linguistics and 22nd annual meeting on Association for Computational Linguistics.* Stroudsburg, PA, USA, http://dl.acm.org/citation.cfm?id=980549.

Gross, M. (1986). Lexicon-Grammar. The representation of compound words. *Proceedings of COLING '86*, Bonn, University of Bonn, http://acl.ldc.upenn.edu/C/C86/C86-1001.pdf.

Gross, M. (1988). Methods and Tactics in the Construction of a Lexicon-Grammar. In *Linguistics in the Morning Calm* 2, Selected Papers from SICOL 1986, pp. 177-197, Séoul: Hanshin Pub. Co.

Gross, M. (1989). La construction de dictionnaires électroniques. *Annales des Télécommunications*, vol. 44, n° 1-2: 4-19, CENT, Issy-les-Moulineaux/Lannion.

Gross, M., Halle M. & Schützenberger M.P. (1973). Formal analysis of natural languages. *Proceedings of the first international conference* (Paris 1970). Paris: The Hague.

Gross, M., Lentin A. (1970). Introduction to Formal Grammars. Berlin/New York: Springer Verlag.

Handschuh, S., Staab S., Ciravegna F. (2002). S-CREAM – Semi-automatic CREAtion of Metadata. The *13th International Conference on Knowledge Engineering and Management (EKAW 2002)*, ed. Gomez-Perez A., Springer Verlag.

Harris, Zellig S. (1946). From Morpheme to Utterance. *Language* 22:3.161–183.

Harris, Zellig S. (1951). *Methods in Structural Linguistics*. Chicago: University of Chicago Press.

Harris, Zellig S. (1952c). Discourse Analysis: A sample text. *Language* 28:4.474-494. (Repr. in 1970a, pp. 349–379.)

Harris, Zellig S. (1954.) Distributional Structure. *Word* 10:2/3.146-162. (Also in *Linguistics Today: Published on the occasion of the Columbia University Bicentennial* ed. by Andre Martinet & Uriel Weinreich, 26-42. New York: Linguistic Circle of New York, 1954.

Harris, Zellig S. (1957). Co-occurrence and transformation in linguistic structure. *Language* 33, pp. 293-340.

Harris, Zellig S. (1962). *String Analysis of Sentence Structure*. Mouton: The Hague.

Harris, Zellig S. (1964). Transformations in Linguistic Structure. *Proceedings of the American Philosophical Society* 108:5, pp. 418-122.

Harris, Zellig S. (1968). Mathematical Structures of Language. *Interscience Tracts in Pure and Applied Mathematics*, 21. New York: Interscience Publishers John Wiley & Sons.

Harris, Zellig S. (1970). *Papers in Structural and Transformational Linguistics*. Dordrecht/ Holland: D. Reidel.

Harris, Zellig S. (1976). *Notes du Cours de Syntaxe*. Transl. and presented by Maurice Gross. Paris: Éditions du Seuil.

Harris, Zellig S. (1981). Papers on Syntax. Hiż H. (ed.). *Synthese Language Library*, 14. Dordrecht/Holland: D. Reidel.

Harris, Zellig S. (1982). *A Grammar of English on Mathematical Principles*. New York: John Wiley & Sons.

Harris, Zellig S. (1988). Language and Information. *Bampton Lectures* *in America*, 28. New York: Columbia University Press.

Harris, Zellig S. (1991). *A Theory of Language and Information: A Mathematical Approach*. Oxford & New York: Clarendon Press.

Harris, Zellig S., Gottfried M., Ryckman T., Mattick P. Jr., Daladier A., Harris T.N. & Harris S. (1989). The Form of Information in Science: Analysis of an immunology sublanguage. *Boston Studies in the Philosophy of Science*, 104. Dordrecht/Holland & Boston: Kluwer Academic Publishers.

Hofstadter, Douglas R. (1999 [1979]). *Gödel, Escher, Bach: An Eternal Golden Braid*. Basic Books.

Horvath, J.A. (2000-2001). Working with Tacit Knowledge. *The Knowledge Management Yearbook*

Kahan, J., Koivunen M., Prud'Hommeaux E., Swick R. (2003). Annotea: An Open RDF Infrastructure for Shared Web Annotations. In *The WWW10 Conference*, Hong Kong, May, pp. 623-632.

Kipper-Schuler, K., Korhonen A., Ryant N., Palmer M. (2006). Extending VerbNet with Novel Verb Classes. *Proceedings of the International Conference on Language Resources and Evaluation* (LREC), Genoa.

Kiryakov, A., Popov B., Kirilov A., Manov D., Ognyanoff D., Goranov M. (2003). Semantic Annotation, Indexing, and Retrieval. In Proceedings of 2nd International Semantic Web Conference (ISWC2003), 20-23 October 2003, Florida, USA.

Kiryakov, A., Simov K.Iv., Ognyanov D. (2002). *Ontology Middleware: Analysis and Design* Del. 38, On-To-Knowledge, March 2002. http://www.ontoknowledge.org/downl/del38.pdf

Knight, K., Marcu, D. (2005). Machine translation in the year 2004. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 5, v/965-v/968.

Lancashire, I. (2000). Corpus linguistics, the humanities, and virtual organizations. Academia/Industry Working Conference on Research Challenges, 49-54.

Laporte, E. & Stavroula V.. (2008). An electronic dictionary of French multiword adverbs. *Proceedings of the LREC Workshop - Towards a Shared Task for Multiword Expressions (MWE 2008).* LREC p. 31-34.

Levelt, Willem J.M. (2008). *An Introduction to the Theory of Formal Languages and Automata*. Nijmegen: Max Planck Institute for Psycholinguistics.

Luna, R. (2011). Così ho regalato il web al mondo. “La Repubblica” 14 novembre 2011. (also on line http://www.repubblica.it/tecnologia/2011/11/14/news/intervista_berners_lee-24969134/?ref=HRERO-1).

Manning, Christopher D. and Schütze H. (1999). *Foundations of Statistical Natural Language Processing*. Massachusetts, London, England: The MIT Press Cambridge.

Manning, Christopher D., Raghavan P. and Schütze H. (2008). Introduction to Information Retrieval. Cambridge University Press. (also online on http://nlp.stanford.edu/IR-book/).

Manov, D, Kiryakov A., Popov B., Bontcheva K., Maynard D., Cunningham H. (2003). Experiments with geographic knowledge for information extraction. *NAACL-HLT 2003, Canada*. *Workshop on the Analysis of Geographic References*, May 31 2003, Edmonton, Alberta.

Martinez, W. (2003). *Contribution à une méthodologie de líanalyse des cooccurrences lexicales multiples dans les corpus textuels,* (Thèse de doctorat) Université de Paris 3.

Maynard, D. ,Tablan D., Ursu C., Cunningham H., Wilks Y. (2001) Named Entity Recognition from Diverse Text Types. *RANLP 2001 Conference*, Tzigov Chark, Bulgaria.

Maynard, D., Tablan V., Bontcheva K., Cunningham H., and Wilks Y. (2003). MUlti-Source Entity recognition – an Information Extraction System for Diverse Text Types. *Technical report CS--02--03*, Univ. of Sheffield, Dep. of CS. http://gate.ac.uk/gate/doc/papers.html.

Miles, A., Brickley, D. (2005). Skos core vocabulary specification. *Technical report, W3C Working Draft.*

Moldovan, D., Mihalcea R. (2001). Document Indexing Using Named Entities. In *Studies in Informatics and Control*, Vol. 10, No. 1, March 2001.

Monteleone, M. (2002). Lessicografia e dizionari elettronici. Dagli usi linguistici alle basi di dati lessicali. Napoli: Fiorentino & New Technology.

Monti, J., Barreiro A., Elia A., Marano F., Napoli A. (2011). Taking on new challenges in multi-word unit processing for Machine Translation. In F. Sanchez-Martinez, J.A. Perez-Ortiz (eds.) Proceedings of the Second International Workshop on Free/Open-Source Rule-Based Machine Translation. Universitat Oberta de Catalunya, Barcelona Spain, 20-21 January 2011, Barcelona: UOC:EDU, pp. 9-11.

Monti, J., Elia A., Postiglione A., Monteleone M., Marano F. (2012 printing) In search of knowledge: text mining dedicated to technical translation. *Proceedings of ASLIB 2011 Translating and the Computer Conference 2011*, 17th-18th November 2011. The Hatton, London.

Nanduri, S., Rugaber S. (1995). Requirements validation via automated natural language parsing. *Proceedings of the Twenty-Eighth Hawaii International Conference on System Sciences*, 3, 362-368.

Nonaka, I. (1994). Theory of Organizational Knowledge Creation. *Organizational Science*, vol 5, no.1.

Nonaka, I., Takeuchi H. (1995). *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*. New York: Oxford University Press.

Parankusham, K.K., Madupu R.R. (2006). *Role of metadata in the datawarehousing environment*. University essay from Luleå/Business Administration and Social Sciences.

Pinker, S. (1996). L'istinto del linguaggio. Milano: Mondadori.

Popov, B., Kiryakov A., Kirilov A., Manov D., Ognyanoff D., Goranov M. (2003). KIM – Semantic Annotation Platform. *The Semantic Web, ISWC, Lecture Notes in Computer Science*, 2003, Volume 2870/2003, 834-849.

Pustejovsky, J., Boguraev B., Verhagen M., Buitelaar P., and Johnston M. (1997). Semantic Indexing and Typed hyperlinking. In *Proceedings of the AAAI Conference, Spring Symposium, NLP for WWW*, 120-128. Stanford University, CA.

Robertson, J. (2007). *There are no KM Systems.* Step Two Designs. Retrieved February 2011 from http://www.steptwo.com.au/papers/cmb_kmsystems/index.html.

Rosenbaum, Peter S. (1967). Phrase structure principles of English complex sentence formation. *Journal of Linguistics,* 3 Vol. 55, No. 4, pp. 859-885.

Sag, I.A., Baldwin T., Bond F., Copestake A. and Flickinger D. (2001). Multiword Expressions: A Pain in the Neck for NLP. *Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics,* CICLing-2002, pages 1-15, Mexico City, Mexico.

Schmidt, T. Wörner K. (2005). Erstellen und Analysieren von Gesprächskorpora mit EXMARaLDA. *Gesprächsforschung* (6) 171-195.

Schumacher, H. (1988). *Valenzbibliografie, Institut für Deutsche Sprache*. Mannheim.

Sekine, S., Sudo K., Nobata C. (2002). Extended Named Entity Hierarchy. *Proceedings of the Language Resource and Evaluation Conference*.

Senge, Peter M., Kleiner A., Roberts C., Ross R.B., Smith B.J. (1994). *The Fifth Discipline.* Fieldbook New York: Currency Doubleday.

Silberztein, M. (1993). *Dictionnaires électroniques et analyse automatique de textes*. Paris: Masson.

Silberztein, M. (2002). *NooJ Manual*. Available for download at: www.nooj4nlp.net.

Sinclair, J. (1991). *Corpus, Concordance, Collocation*. New York: Oxford University Press.

Tesnière, L. (1953). Esquisse d'une syntaxe structurale. Klincksieck, Paris.

Tesnière, L. (1959). Éléments de syntaxe structurale. Klincksieck, Paris.

Thierauf, R. J. (1999). *Knowledge Management Systems.* Quorum Books.

Toffler, A. (1970). *Future Shock*. USA: Random House.

Turing, Alan M. (1936). On Computable Numbers, with an Application to the Entscheidungs problem. *Proceedings of the London Mathematical Society*.

Turing, Alan M. (1948). Intelligent Machinery. Reprinted in *Cybernetics: Key Papers*. Ed. C.R. Evans and A.D.J. Robertson. Baltimore: University Park Press, 1968. p. 31.

Van Guilder, L. (1995). Automated part of speech tagging: a brief overview. *Handout for LING 361*, Georgetown University.

Vargas-Vera, M., Motta E., Domingue J., Lanzoni M., Stutt A. and Ciravegna F. (2002). MnM: Ontology Driven Semi-Automatic and Automatic Support for Semantic Markup. In *Proceedings of EKAW 2002*, ed. Gomez-

Perez A., Springer Verlag.

Vietri, S. (2001). *Navigare nei testi. Teorie e applicazioni informatiche per la linguistica testuale*. Napoli: Editoriale Scientifica Italiana.

Vietri, S. (2004). *Lessico-grammatica dell'italiano. Metodi, descrizioni, applicazioni*. Torino: UTET.

Vietri, S. (2008). *Dizionari elettronici e grammatiche a stati finiti. Metodi di analisi formale della lingua italiana*. Salerno: Plectica.

Vietri, S., Elia A., D'Agostino E. (2004). Lexicon-grammar, Electronic Dictionaries and Local Grammars in Italian. Laporte, Leclère, C., Piot, M., Silberztein M. (eds.), *Syntaxe, Lexique et Lexique-Grammaire. Volume dédié à Maurice Gross, Lingvisticae Investigationes Supplementa 24*. Amsterdam/ Philadelphia: John Benjamins.

Wagner, A. Zeisler, B. (2004). A syntactically annotated corpus of Tibetan. In *Proceeding Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, Lisboa, Portugal, May 2004.

Wickramasinghe, N., Bali R.K., Lehaney B., Schaffer J.L., Gibbons M.C. (2009). *Healthcare Knowledge Management Primer*. New York & London: Routledge Taylor & Francis Group.

Wickramasinghe, N., Gupta J.N.D, Sharma S.K. (2005). *Creating Knowledge-Based Healthcare Organizations*, Idea Group Publishing, Hershey.

Yule, G.U. (1944). *A statistical study of vocabulary*. Cambridge: Cambridge University Press.

Zipf, G.K. (1935). *The psychobiology of language - An introduction to dynamic philology.* Boston: Houghton-Mifflin.