Institute of Informatics & Telecommunications

Software & Knowledge Engineering Laboratory

Ellogon Components' Specifications

Stergos D. Afantenos George Petasis Vangelis Karkaletsis

June 2002

Table of Contents

<u>1</u> INTRODUCTION	2
2 GREEK TOKENIZER SPLITTER	3
2.1 INPUT2.2 OUTPUT	3 3
<u>3</u> HELLENIC POS TAGGER	6
3.1 INPUT3.2 OUTPUT	6 6
4 UNICODE LIST LOOKUP	9
4.1 INPUT4.2 OUTPUT	9 10
5 ADD SENTENCE CONSTITUENTS	11
5.1 INPUT 5.2 OUTPUT	11 11

1 Introduction

This document is intended to accompany the *Ellogon* text engineering platform, along with two other documents: the *Users' Guide to Ellogon* and the *Developers' Guide to Ellogon*. In those manuals we have described the *Ellogon* platform in detail, and we have furthermore described how to build and run Components and Systems for *Ellogon*, among other things.

The current distribution of *Ellogon* comes equipped with some predefined Components. The purpose of this document is to describe those Components and what they are doing.

2 Greek Tokenizer Splitter

The aim of the Greek Tokenizer Splitter Component is to identify all the tokens inside a text and annotate them properly. Tokens are all the words which are separated between them with blank spaces or punctuation marks. Additionally, all the punctuation marks are considered to be tokens.

Furthermore, this Component also tries to identify sentence boundaries. In order to achieve this, it uses a simple algorithm based mainly on punctuation marks and capitalization.

2.1 Input

The input to the Greek Tokenizer Splitter is just a Collection of Documents which need not have any other annotations, from another Component. In other words, this module has no pre-conditions. The only constraint that this component poses is for the Documents to be written in the Greek language.

2.2 Output

The Greek Tokenizer Splitter tries to identify all tokens of a Document and annotate them appropriately according to their type. By the type of a token we mean an indication on whether it is, for example, a punctuation mark, an English uppercase word, a Greek lower case word, *etc.* In other words, this Component places annotations of type token for all the token it identifies with an attribute type according to the type of the token. You can see all the types of attributes that this Component places in *Table 2.1*. In *Figure 2.1* you can see an example of a token annotation.

This Component also tries to identify all the sentences in the Document. In order to do so, it uses a simple algorithm which is based on punctuation marks and capitalizations. For every sentence it finds it creates a new annotation of type sentence with attribute constituents. The constituents contain the ids of all the tokens which constitute the sentence. An example of a sentence annotation you can see in *Figure 2.2*.

💽 Konsole – Konsole 🔹	Ellogon v 1.0: Inte 🍳	System: tem	p	• System: Greek S	٩
$-$ - M Explore Annotations of Document econ_gre.txt, From Collection econ_gre $$ $$ $$ $$ $$ $$ $$ $$ $$ $$					
🛛 🖶 🛷 🙊 🔍 🖆	🕱 🖾 📔 🦸				
Νέες μεγάλες απώλειες κατα συνεδριάσεων της Παρασκευ κεφαλαιαγορές. Παρά τα ήδη βρίσκονται οι τιμές των περισ επιχειρηματικών ειδήσεων αλ αξιολόγησης οδήγησε σε υπο Όπως σχολιάζουν παράγοντε συνεχόμενη εβδομάδα απωλε επιχειρήσεων να πείσουν ότι διατηρήσουν την κερδοφορία αράλληλα, σημειώνεται ότι α είναι δυσκολότερο να αντιμε προκαλούνται από συγκεκριμα	ίς οι μεγάλες έυρώπαϊκέ ιδιαίτερα χαμηλά επίπεδα σότερων μετοχών, μία σε λά και υποβαθμίσεων από χώρηση πολλούς κλάδου ς των κεφαλαιαγορών, η ιών είναι απόρροια της αἰ πορούν να αυξήσουν ή ς τους την παρούσα περίοζ γτού του είδους η ύφεση τωπιστεί, από αντίστοιχες να γεγονότα.	ς ίστα οποία οίκους ς. τέταρτη δυναμίας των ικόμη και να ίο. Π στην αγορά		<mark>token -</mark> sentence +	
	Туре	Start	End	Attributes	<u> </u>
0	token	0	4	type=GFW	
1	token	5	12	type=GLW	
2	token	13	21	type=GLW	
<u>2</u> 3	token	22	32	type=GLW	
4	token	33	36	type=GLW	
5	token	37	40	type=GLW	
6	token	41	45	type=GLW	
7	token	46	49	type=GLW	
8	token	50	62	type=GLW	
9	token	63	66	type=GLW	
10	token	67	77	type=GFW	
11	token	78	80	type=GLW	∇
110	• 1	01		1 0111	2

Figure 2.1 An Example of a Token Annotation



Figure 2.2 An Example of Sentence Annotation

Token Type	Token Description
GLW	Only Greek lower case characters are present in the token (α-ω, e.g. "λέξη").
GUW	Only Greek upper case characters are present in the token (A- Ω , e.g. "AEEH").
GFW	Only Greek characters comprise the token. The first is an upper case character and the rest are lower case characters (e.g. "Λέξη").
GW	Only Greek characters are present in the token and the token does not belong to the GLW, or GUW, or GFW types (e.g. " λ E ξ H").
ELW	Only lower case, Latin characters are present in the token (a-z, e.g. "word").
EUW	Only upper case Latin characters comprise the token (A-Z, e.g. "WORD").
EFW	Only Latin characters comprise the token. The first character is upper case and the rest are lower case (e.g. "Word").
EW	The token is comprised of Latin characters only and it does not be- long to any of the ELW, or EUW or EFW types (e.g. "wOrD").
MLW	The token is comprised of Greek lower case and Latin lower case characters. It does not contain characters of any other kind (e.g. "ναπαtel").
MUW	The token is comprised of Greek upper case characters and Latin uppercase characters. The token does not contain characters of any other kind (e.g. "KANATEL").
MFW	The token is comprised of Greek and Latin alphabetic characters. It does not contain characters of any other kind. The first character is uppercase and the rest are lower case (e.g. "Kαπαtel").
MW	The token is comprised of Greek and Latin characters and it is not classified as MLW or MUW or MFW (e.g. "μαπαTEL").
NUM	The token is comprised of numeric characters exclusively (0-9, e.g. "1999").
WNUM	The token is comprised of one or more numeric characters and Gree or Latin or both alphabetic characters The token does not belong to any of the aforementioned categories (e.g A1).
PUNCTUATION	The token is one of the following characters: ! . , ? ;
SYMBOL	The token does not belong to any of the aforementioned categories.

Table 2.1 Types of Attributes for the Tokens

3 Hellenic POS Tagger

The Hellenic POS Tagger tries to identify the part of speech for every token in the Document. This component is based on code written by Eric Brill in 1995¹ and was later modified by George Petasis in 1999² in order to find Greek parts of speech.

3.1 Input

The input to this Component are Documents written in the Greek language. Also this component presupposes that the Greek Tokenizer Splitter Component has been run against the Collection. In other words it expects to find token and sentence Annotations in the Documents.

3.2 Output

The result of the processing of this Component is simply to add an additional attribute to every token. The attribute is of type pos and it indicates what part of speech this token is. If, for some reason, it fails to find a part of speech for a specific token, then the attribute pos is empty. An example you can see in *Figure 3.1*. In the Table of the following page you can see all the possible values of the pos attribute.

¹ Brill, E., "Transformation-Based Error-Driven Learning and Natural Language Processing: A CaseStudy in Part of Speech Tagging", Computational Linguistics, vol. 21, n. 24, 1995.

² G. Petasis, G. Paliouras, V. Karkaletsis, C.D. Spyropoulos and I. Androutsopoulos, "Using Machine Learning Techniques for Part-of-Speech Tagging in the Greek Language", *Proceedings of the 7th Hellenic Conference on Informatics*, Ioannina, Greece, 1999.

			C	ategori	es		
	Articles			0			
DDT	Definite	Article		IDT	Indefinite Article		
	Nouns						
	PoS	Number	Gen- der		Pos	Number	Gende
NNM	Noun	Singular	Male	NNPM	Proper Name	Singular	Male
NNF	Noun	Singular	Female	NNPF	Proper Name	Singular	Female
NNN	Noun	Singular	Neuter	NNPN	Proper Name	Singular	Neuter
NNSM	Noun	Plural	Male	NNPSM	Proper Name	Plural	Male
NNSF	Noun	Plural	Female	NNPSF	Proper Name	Plural	Female
NNSN	Noun	Plural	Neuter	NNPSN	Proper Name	Plural	Neuter
	Adjectiv						
	PoS	Number	Gen- der		Pos	Number	Gen- der
JJM	Adjec- tive	Singular	Male	JJSM	Adjective	Plural	Male
JJF	Adjec- tive	Singular	Female	JJSF	Adjective	Plural	Fema
JJN	Adjec- tive	Singular	Neuter	JJSN	Adjective	Plural	Neute
CD	Cardina	l Numerals (e	e.g. ένα (on	ne), δύο (two), τϱἰα (three), .	and numbers: 1,	2, 3)
	Pronou	ns					
PRP		l Pronoun		IP	Demonstrati		
РР		ve Pronoun		WP	Relative Pronoun		
REP		e Pronoun		QP	Interrogative		
DP		Pronoun		INP	Indefinite Pr	onoun	
TE	Verbs						
VB		Tense Verb		VBS		se Verb, Plural	
VBD		nse Verb		VBDS	Past Tense V		
VBF		Tense Verb		VBFS	Future Tens	e Verb, Plural	
MD	Particip	/erb (έχω (ha	ινε), ειμαι ((be))			
VBG	-	oice Particle	<u>,</u>	VBPD	Past Particip	10	
VBG VBP		Participle	·	VDID	1 aoi 1 atucip		
1.01		Γάτισιριε Μέρη του Λο	່ານດາ				
RB	Adverb	and the main and the	100	RP	Particle		
IN	Preposit	ion		UH	Expletive/Exclamation		
CC	Conjune			FW	Non-Greek		
		ymbols		· ·	Tion Oreck		
DATE	Date	<u>,</u>		:	Colon (:)		
TIME	Time			;	Semi Colon (i.e. Greek Question Mark) (; ?)		
AB	Abbreviation			!	Exclamation	Mark (!)	
SYM	Symbol			(Left Bracket		
	Full stop	o (.))	Right Bracke		
,	Comma			"		on Mark (", «) and	l Right

ο 👧 🚸 🖫	. 🧟 🗗 🗟 🛛		re.txt, From Collecti	on econ_gre 👘 👘 🗌
ιέες μεγάλες <mark>απ</mark>		¥ [🚛		
Ιέες μεγάλες <mark>απ</mark> ης Παρασκευής (8 🏸		
διάιτερα χαμηλά ιερισσότερων μετ ποβαθμίσεων απι ιολλούς κλάδους Οπως σχολιάζους τυνεχάμενη εβδο πιχειρήσεων να r ιατηρήσουν την ι ημειώνεται ότι α	οι μεγάλες εύρωπα snineča στα onoia roxών, μία σειρά επ ό οίκους αξιολόγης 3. / παράγοντες των ι μάδα απωλειών είν πείσουν ότι μπορού κερδοφορία τους τ υτού του είδους η αντιμετωπιστεί, απά	ϊκές κεφαλαία βρίσκονται οι τ ιχειρηματικών της οδήγησε σ «εφαλαιαγορώ αι απόρροια τη ν να αυξήσουν ήν παρούσα πε ύφεση στην αγ	είδήσεων αλλά και ε υποχώρηση ης αδυναμίας των ν ή ακόμη και να ερίοδο. Παράλληλα, γορά είναι	Sentence +
			A T	
	Туре	Start	End	Attributes
-	token	0	4	type=GFW, pos=JJSF
	token	5	12	type=GLW, pos=JJSF
	token	13	21	type=GLW, pos=NNSF
·	token	22	32	type=GLW, pos=VBDS
	token	33	36	type=GLW, pos=IN
	token	37	40	type=GLW, pos=DDT
	token	41	45	type=GLW, pos=NNF
	token	46	49	type=GLW, pos=DDT
	token	50	62	type=GLW, pos=NNSF
	token	63	66	type=GLW, pos=DDT
0	token	67	77	type=GFW, pos=NNPM
-		70	00	1 07.77 DDM

Figure 3.1An Example of a Part of Speech Attribute

4 Unicode List Lookup

The function of the Unicode List Lookup Component is to compare the tokens (see Chapter 2) in the Documents of a Collection against a set of lists. For each of the elements in the lists if it matches a token, a new annotation is created, with the same spans as the token. The type of the annotation is lookup and the attribute is the name of the file which constitutes the list.

This Component is especially helpful, for example, in cases you have some lists with named entities, such as persons' names, locations' names, organizations' names, *etc.*

4.1 Input

The input to this component is a Collection which has run against a tokenizer, for example the Greek Tokenizer splitter described in Chapter 2. Additionally, this component needs a set of lists which you can give as input through the parameters of the Component. Note that in the present distribution of Ellogon, there are no lists and the Ellogon user should develop his own. In *Figure 4.1* you can see the parameters of the Component. in order to add your own lists push the "Create Rule File". Additionally, in this dialog you can specify the encoding of your list files, and the rule file (see below).

👿 Shell - Korsole 🔹	Ellogon v 1.0: Inte • System: temp • Module Param	eter				
	UnicodeListLookup	$\cdot \Box \times$				
Module Paramet	ers UnicodeListLookup					
Author: petasis, Wed Nov 21 22:01:12 EET 2001 Part of the Ellogon Text Engineering Platform						
Lookup Rule File:	pplications/Ellogon/share/modules/UnicodeListLookup/Rules.txt	<i>≌</i>				
Lookup Rule File Encoding: 📩 📩 iso8859-7 🦟						
	Create Rule File					
Close Apply Default						

Figure 4.1 The Unicode List Lookup Module's Parameters

In *Figure 4.2* you can see the dialog that appears. With this dialog you can specify the list files that you want by pressing the associated Browse button. Note that the name of your list files should have the extension .lst. Also, each line of your list file should contain only one element.

Apart from the list files, you can also define a name for the rule file. This rule file will be created by the Component, and will be based on the list files you have specified. *Ellogon* provides a default name for that file.

💽 Shell - Konsole	 Ellogon v 1.0: Inte 	 System: temp 	٩
	kup Rule File	·	\times
Select List Files:			
k2b/applications/Ellogo	n/share/modules/Unicodel	<u>.istLookup/Lists</u>	
		🛱	.
Select Rule File:			_
applications/Ellogon/sh	are/modules/UnicodeListL	ookup/Rules.txt 🛛 😅	•
1-			
	Ok Cancel		

Figure 4.2 Defining your own lists

Once you have finished with defining the list files push the OK button

4.2 Output

This Component tries to match the elements of all the lists with the tokens of the Documents in the Collection. For every token that matches an element of a list with the name my_list.lst it creates a new annotation of type lookup and attribute my_list.

5 Add Sentence Constituents

The function of the Add Sentence Constituents Component is to calculate the constituents, *i.e.* the ids of the tokens that constitute a sentence, and place them as an attribute in the sentence annotation.

5.1 Input

The Add Sentence Constituents Component takes as input Documents in a Collection which contain sentence annotations. Usually those annotations will also have as attribute a set of constituents which contain the ids of the tokens which constitute the sentence.

5.2 Output

This component discards all the constituents of the sentence annotation and recalculates its constituents. Then it places the ids of the tokens that it has calculated that constitute the sentence as the new constituents attribute.

This Component is especially useful in cases where some tokens have been deleted from the Documents. In such a case you can "correct" the sentence constituents with this Component.