

A Question Answering system with semantic analysis based on the ER model

Σύστημα Ερωταποκρίσεων σημασιολογικής ανάλυσης με βάση το ER Model

ΙΩΑΝΝΗΣ ΤΣΑΜΠΟΣ

Μεταπτυχιακή Διατριβή

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ

Μάρτιος 2018

Επιβλέπων Καθηγητής:
Εμμανουήλ Μαρακάκης

Δήλωση αυθεντικότητας

Το έργο που περιέχεται σε αυτή τη διατριβή δεν έχει υποβληθεί προηγουμένως για πτυχίο ή δίπλωμα σε οποιοδήποτε άλλο ίδρυμα τριτοβάθμιας εκπαίδευσης ή για οποιονδήποτε άλλο σκοπό. Από όσο γνωρίζω και πιστεύω, η διατριβή δεν περιέχει υλικό που έχει δημοσιευθεί ή συνταχθεί από άλλο πρόσωπο εκτός από τις περιπτώσεις που αναφέρονται στις αναφορές. Βεβαιώνω ότι το πνευματικό περιεχόμενο αυτής της διατριβής είναι το προϊόν της δικής μου δουλειάς και έχει αναγνωριστεί όλη η βοήθεια που παρασχέθηκε στην προετοιμασία αυτής της εργασίας καθώς και οι πηγές.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα της διατριβής μου Καθηγητή Δρ. Εμμανουήλ Μαρακάκη, για την πολύτιμη βοήθεια και καθοδήγησή του. Επιπλέον θα ήθελα να ευχαριστήσω για την καθοδήγηση και τη συμμετοχή του τον Δρ. Χαρίδημο Κονδυλάκη. Ευχαριστώ επίσης την οικογένειά μου η οποία με την υποστήριξη και την υπομονή της συνέβαλε καθοριστικά στην ολοκλήρωση αυτής της μεταπτυχιακής διατριβής.

Περίληψη

Τα συστήματα ερωταποκρίσεων αποτελούν την ιδανική λύση στην ανάγκη του σύγχρονου χρήστη των υπολογιστών και των έξυπνων συσκευών προκειμένου να αποκτήσουν τις ζητούμενες πληροφορίες με ακριβή τρόπο. Μπορούν να προσφέρουν στοχευμένες, πλήρεις και σαφείς απαντήσεις στα ερωτήματα χωρίς περιττές πληροφορίες και χωρίς να χρειάζεται η αναζήτηση από μέρος του της απάντησης μέσα στις σελίδες που επιστρέφει μια μηχανή αναζήτησης. Για να είναι ένα σύστημα σε θέση να ανταποκριθεί δίνοντας τη ζητούμενη απάντηση στα ερωτήματα των χρηστών, θα πρέπει να έχει τη δυνατότητα να κατανοήσει τις έννοιες που περιέχονται μέσα στα έγγραφα, όσο και τις έννοιες που εμπεριέχουν τα ερωτήματα των χρηστών.

Στα πλαίσια της μεταπτυχιακής διατριβής μελετήθηκε, σχεδιάστηκε και αναπτύχθηκε ένα σύστημα ερωταποκρίσεων το οποίο αναλύει και αναπαριστά σημασιολογικά κείμενα γραμμένα στην Ελληνική γλώσσα. Η ανάλυση των κειμένων γίνεται σε λεξικολογικό, γραμματικό, συντακτικό και σημασιολογικό επίπεδο. Η αναπαράσταση σημασιολογικής δομής των κειμένων η οποία βασίζεται στο μοντέλο Οντοτήτων – Συσχετίσεων αποθηκεύεται στη βάση γνώσης του συστήματος. Με τον ίδιο τρόπο αναλύονται και αποθηκεύονται στη βάση γνώσης οι σημασιολογικές δομές των ερωτημάτων των χρηστών. Το σύστημά μας είναι σε θέση να δεχτεί συγκεκριμένου τύπου ερωτήματα που αφορούν στο περιεχόμενο των κειμένων, να αναζητήσει τις σχετικές πληροφορίες και να συνθέσει τις σωστές απαντήσεις σε φυσική γλώσσα, οι οποίες εμφανίζονται ως απόκριση του συστήματος προς τον χρήστη.

Abstract

Question Answering Systems offer an ideal solution to the needs of users of modern computers and smart devices in order to get the requested information in an accurate way. They can offer targeted, complete and clear answers to users' questions without unnecessary information and without having the users to search for the right answer within the pages which are returned by a search engine. A Question Answering system should be able to understand the concepts contained within documents and the concepts of users' questions in order to be able to respond to the users' requests by providing the required answer.

In the framework of the present thesis, a question answering system has been studied, designed and developed, which analyzes and represents semantically texts written in the Greek language. Input texts are analyzed lexicologically, grammatically, syntactically and semantically. The representation of the structure of texts is stored in the knowledge base of the system. In the same way, the questions of users are analyzed and the created semantic structures are stored in the knowledge base as well. Our system is able to accept specific types of questions related to the content of the input text, to search for the relevant information, and to synthesize the correct response in natural language which is returned to the user.

Πίνακας Περιεχομένων

Δήλωση αυθεντικότητας	i
Ευχαριστίες	iii
Λίστα Σχημάτων	iii
Λίστα Πινάκων	v
1. Επισκόπηση και κίνητρο	1
1.1 Εισαγωγή	1
1.2 Ορισμός του προβλήματος.....	2
1.3 Η δομή της διατριβής.....	2
2. Θεωρητικό υπόβαθρο και σχετικές εργασίες.....	3
2.1 Επεξεργασία φυσικής γλώσσας.....	3
2.2 Αναπαράσταση Γνώσης.....	5
2.3 Συστήματα Ερωταποκρίσεων (Question Answering Systems).....	7
2.4 Μοντέλο Οντοτήτων – Συσχετίσεων (ER Model).....	12
2.5 Επεξεργασία φυσικής γλώσσας για την Ελληνική γλώσσα.....	13
2.6 Συντακτικό της γλώσσας με κανόνες BNF και DCG	14
2.6.1 Περιγραφή γραμματικής με κανόνες BNF (Backus-Naur Form)	14
2.6.2 Περιγραφή γραμματικής με κανόνες DCG (definite clause grammar)	15
2.7 Web-Programming με την SWI-Prolog.....	17
2.8 Σχετική δουλειά	18
2.8.1 Ανάλυση κειμένου από Cloud Natural Language της Google.....	18
2.8.2 Ανάλυση κειμένου από το Stanford NLP	19
2.8.3 Ανάλυση κειμένου από το Ινστιτούτο Επεξεργασίας του Λόγου (ΙΕΛ)	22
3. Περιγραφή του συστήματος ερωταποκρίσεων με τη χρήση εξαγωγής πληροφοριών από κείμενα	23
3.1 Κύρια χαρακτηριστικά του συστήματος μας.....	23
3.2 Περιορισμοί και επεκτάσεις.....	24
4. Αρχιτεκτονική του συστήματος.....	25
5. Αναλυτική περιγραφή του συστήματος	27
5.1 Λεξικολογική, γραμματική και συντακτική ανάλυση	27
5.1.1 Λεξικά	27
5.1.2 Παραγωγή κλίσεων λέξεων	36

5.1.3	Συλλαβισμός λέξεων	44
5.1.4	Τονισμός λέξεων	45
5.1.5	Διαδικασία δημιουργίας δέντρου συντακτικής ανάλυσης	48
5.1.6	Κανόνες γραμματικής της ελληνικής γλώσσας.....	53
5.1.7	Αναπαράσταση της δομής των προτάσεων	56
5.2	Εξαγωγή οντοτήτων και συσχετίσεων	75
5.2.1	Πίνακες συμβόλων	75
5.2.2	Κανόνες του μοντέλου οντοτήτων-συσχετίσεων.....	78
5.3	Αποθήκευση της σημασιολογικής δομής αναπαράστασης ενός κειμένου στη βάση γνώσης..	85
5.4	Απαντήσεις σε ερωτήματα.....	98
5.4.1	Περιγραφή	98
5.4.2	Αλγόριθμος δημιουργίας απάντησης	98
5.4.3	Δομικά κατηγορήματα	101
5.4.4	Παραδείγματα.....	105
5.5	User Interface	110
5.5.1	Περιγραφή	110
5.5.2	Γραφική αναπαράσταση της δομής των προτάσεων.....	110
5.5.3	SWI Prolog Web Server.....	112
5.5.4	HTTP Request handling.....	113
5.5.5	Ενσωμάτωση και εκτέλεση κώδικα στη σελίδα HTML.....	113
5.5.6	Εντολές ελέγχου δεδομένων λεξικών	118
5.6	Χρήση του user interface	118
6.	Παραδείγματα από τη χρήση του συστήματος	121
7.	Μελλοντικές επεκτάσεις.....	129
8.	Συμπεράσματα	131
9.	Βιβλιογραφικές Αναφορές.....	133

Λίστα Σχημάτων

Σχήμα 1. Η συντακτική ανάλυση της πρότασης «Η γραμματική οριστικών προτάσεων αποτελείται από τρία συντακτικά μέρη» από το service του ILSP NLP tools [25]	4
Σχήμα 2. Το δέντρο συντακτικής ανάλυσης (Parse Tree) της πρότασης «Ο σκύλος κυνηγάει τη γάτα»	5
Σχήμα 3: Η αρχιτεκτονική ενός συστήματος ερωταποκρίσεων σε υψηλό επίπεδο (Kolomiyets, 2011) [26]	10
Σχήμα 4: Η Δομή ενός συστήματος ερωταποκρίσεων (Allam, 2012) [2]	12
Σχήμα 5: Η ανάλυση με την εφαρμογή Cloud Natural Language της Google [16] για την πρόταση: “A digital signature is a mathematical scheme for demonstrating the authenticity of digital messages or documents”	19
Σχήμα 6: Η ανάλυση με τον Stanford Parser [48] της πρότασης: “A digital signature is a mathematical scheme for demonstrating the authenticity of digital messages or documents”	21
Σχήμα 7: Η ανάλυση με από την υπηρεσία IEL dependency parser [23] της πρότασης: «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά»	22
Σχήμα 8. Η τοπολογία Client - Server του Συστήματος Ερωταποκρίσεων	25
Σχήμα 9. Τα στάδια ανάλυσης των κειμένων από την εφαρμογή	25
Σχήμα 10. Τα στάδια για τη δημιουργία της απάντησης στο ερώτημα του χρήστη	26
Σχήμα 11. Τα στάδια για τη δημιουργία του δέντρου συντακτικής ανάλυσης	48
Σχήμα 12. Το δέντρο σύνταξης της πρότασης «Ο μαθητής διαβάζει ένα βιβλίο»	52
Σχήμα 13: Η γενική αναπαράσταση της δομής μιας πρότασης	58
Σχήμα 14: Αναπαράσταση δομής ουσιαστικού	59
Σχήμα 15: Αναπαράσταση δομής του επιθετικού προσδιορισμού	60
Σχήμα 16: Αναπαράσταση δομής επιρρήματος	61
Σχήμα 17: Σχηματική αναπαράσταση της δομής της πρότασης «Ο μαθητής διαβάζει ένα βιβλίο»	61
Σχήμα 18: Σχηματική αναπαράσταση της δομής της πρότασης «Το λεμόνι δίνει νόστιμη γεύση στο φαγητό»	63
Σχήμα 19: Σχηματική αναπαράσταση της δομής της πρότασης «Η κύρια είσοδος του ψηλού πολυόροφου κτιρίου ανοίγει αυτόματα»	64
Σχήμα 20: Σχηματική αναπαράσταση της δομής της πρότασης «Ο Νίκος περνά και χαιρετά»	65
Σχήμα 21: Σχηματική αναπαράσταση της δομής της πρότασης «Η κεντρική τράπεζα καθορίζει το μέγιστο ονομαστικό επιτόκιο της εξάμηνης προθεσμιακής κατάθεσης καθημερινά»	66
Σχήμα 22: Σχηματική αναπαράσταση της δομής της πρότασης «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά»	68
Σχήμα 23: Σχηματική αναπαράσταση της δομής της πρότασης «Οι μεγαλύτεροι πλανήτες του ηλιακού συστήματος είναι οι πλανήτες Δίας και Κρόνος»	69
Σχήμα 24: Σχηματική αναπαράσταση της δομής της πρότασης «Το μέγιστο ηλεκτρικό ρεύμα φόρτισης της μπαταρίας του φωτοβολταϊκού συστήματος που προτείνεται ανεπιφύλακτα αναφέρεται στο συνοδευτικό εγχειρίδιο»	72

Σχήμα 25. Τα βήματα για την αποθήκευση στη βάση γνώση της αναπαράστασης της σημασιολογικής δομής της πρότασης «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά».....	89
Σχήμα 26: Η αρχιτεκτονική του συστήματος δημιουργίας απαντήσεων σε ερωτήσεις.....	98
Σχήμα 27. Τα βήματα για την αναζήτηση της απάντησης στο ερώτημα «Τι είναι η ψηφιακή υπογραφή;».....	105
Σχήμα 28. Τα βήματα για την αναζήτηση της απάντησης στο ερώτημα «Ποιος είναι μαθηματικό σύστημα».....	106
Σχήμα 29. Γραφική αναπαράσταση της σημασιολογικής δομής του κειμένου «Ο μαθητής διαβάζει ένα βιβλίο. Η όμορφη Μαρία φοράει ένα κόκκινο κοντό φόρεμα. Ο Γιάννης τρέχει γρήγορα.».....	108
Σχήμα 30. Η τοπολογία του συστήματος ερωταποκρίσεων Knowledge.....	110
Σχήμα 31. Η αναπαράσταση της σημασιολογικής δομής της πρότασης «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά» όπως εμφανίζεται στον χρήστη.....	111
Σχήμα 32. Η αναπαράσταση της σημασιολογικής δομής της πρότασης «Ο καθηγητής Γιώργος διδάσκει μαθηματικά. Ο μαθητής Γιάννης παρακολουθεί μαθηματικά....» όπως εμφανίζεται στο χρήστη.....	112
Σχήμα 33. Η έξοδος της εφαρμογής για το κείμενο «Ο μαθητής διαβάζει ένα βιβλίο».....	116
Σχήμα 34. Η έξοδος της εφαρμογής για το κείμενο «Ο καλός μαθητής διαβάζει ένα μεγάλο βιβλίο γρήγορα».....	118
Σχήμα 35. Τα περιεχόμενα της σελίδας της εφαρμογής για το κείμενο εισόδου: «Ο μαθητής διαβάζει ένα βιβλίο. Η όμορφη Μαρία φοράει ένα κόκκινο κοντό φόρεμα. Ο Γιάννης τρέχει γρήγορα».....	121
Σχήμα 36. Οι απαντήσεις σε διαφορετικά ερωτήματα αφότου δοθεί το κείμενο εισόδου: «Ο μαθητής διαβάζει ένα βιβλίο. Η όμορφη Μαρία φοράει ένα κόκκινο κοντό φόρεμα. Ο Γιάννης τρέχει γρήγορα».....	122
Σχήμα 37. Τα περιεχόμενα της σελίδας της εφαρμογής για το κείμενο εισόδου με ορθογραφικό λάθος: «Ο μαθητής διαβάζει ένα βιβλίο».....	123
Σχήμα 38. Τα περιεχόμενα της σελίδας της εφαρμογής για το κείμενο εισόδου: «Η όμορφη ψηλή γυναίκα του νέου μανάβη της γειτονιάς που περνά τώρα φοράει μοντέρνο πράσινο φόρεμα».....	124
Σχήμα 39. Οι απαντήσεις σε διαφορετικά ερωτήματα αφότου δοθεί το κείμενο εισόδου: «Η όμορφη ψηλή γυναίκα του νέου μανάβη της γειτονιάς που περνά τώρα φοράει μοντέρνο πράσινο φόρεμα».....	125
Σχήμα 40. Τα περιεχόμενα της σελίδας της εφαρμογής για το κείμενο εισόδου: «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά. Η ψηφιακή υπογραφή πιστοποιεί την προέλευση του εγγράφου και πλαστογραφείται δύσκολα».....	126
Σχήμα 41. Οι απαντήσεις σε διαφορετικά ερωτήματα αφότου δοθεί το κείμενο εισόδου: «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά. Η ψηφιακή υπογραφή πιστοποιεί την προέλευση του εγγράφου και πλαστογραφείται δύσκολα».....	126

Λίστα Πινάκων

Πίνακας 1. Οι δημοφιλέστερες σχολές στην αναπαράσταση γνώσης και οι αντίστοιχες αναφορές στον τομέα της φυσικής γλώσσας [11]	7
Πίνακας 2. Κατηγοριοποίηση των συστημάτων ερωταποκρίσεων [43]	9
Πίνακας 3. Τα μέρη του λόγου και τα αντίστοιχα αρχεία Prolog με τα λεξικά που τα περιέχουν.....	28
Πίνακας 4. Οι καταλήξεις για τα μέρη του λόγου και τα αντίστοιχα αρχεία Prolog τις περιέχουν	28
Πίνακας 5. Η κλίση του ουσιαστικού μαθητής όπως παράγεται από το κατηγορήμα mk_gr_noun/5.....	38
Πίνακας 6. Η κλίση του επιθέτου «ψηφιακός» όπως παράγεται από το κατηγορήμα mk_gr_adj/5	41
Πίνακας 7. Η κλίση του ρήματος «διαθέτω» όπως παράγεται από το κατηγορήμα mk_gr_verb/5	43
Πίνακας 8. Ο αριθμός συλλαβών για κάθε λέξη που υπολογίζεται από το κατηγορήμα spellCount/2....	45
Πίνακας 9. Τα σχήματα αναπαράστασης κάθε στοιχείου του μοντέλου οντοτήτων-συσχετίσεων.....	76
Πίνακας 10. Η αναπαράσταση των συνδέσεων μεταξύ των στοιχείων του μοντέλου οντοτήτων-συσχετίσεων και η σημασία τους	77
Πίνακας 11. Οι κανόνες δημιουργίας των στοιχείων του Μοντέλου Οντοτήτων – Συσχετίσεων από τη γραμματική των προτάσεων.....	85
Πίνακας 12. Οι τιμές του RefTable και το αντίστοιχο κατηγορήμα αναφοράς	87
Πίνακας 13. Τα γεγονότα του κατηγορήματος relModel/8 σε μορφή πίνακα τα οποία παριστούν τη σημασιολογική δομή του κειμένου του παραδείγματος.....	97
Πίνακας 14. Πίνακας με τις τιμές των ορισμάτων τριών γεγονότων του κατηγορήματος relModel/3 τα οποία δημιουργούνται με την αναπαράσταση της σημασιολογικής δομής της πρότασης «Η δασκάλα διδάσκει ιστορία»	99
Πίνακας 15. Πίνακας με τις τιμές των ορισμάτων όλων των γεγονότων του κατηγορήματος relModel/3 τα οποία δημιουργούνται με την αναπαράσταση της σημασιολογικής δομής της πρότασης «Η δασκάλα διδάσκει ιστορία» και της ερώτησης «Ποιος διδάσκει ιστορία;»	99
Πίνακας 16. Οι τιμές των 5 πρώτων ορισμάτων των γεγονότων του RelModel/8 με τη σημασιολογική αναπαράσταση της ερώτησης «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά».....	102
Πίνακας 17. Τα χαρακτηριστικά της απεικόνισης των στοιχείων του Μοντέλου Οντοτήτων – Συσχετίσεων όπως εμφανίζονται στον χρήστη.	115

1. Επισκόπηση και κίνητρο

Τα συστήματα ερωταποκρίσεων αποτελούν μια τεχνολογική πρόκληση καθώς οι εξελιγμένες δυνατότητες των κινητών συσκευών σε ταχύτητα και user interface, η χρήση τους από μεγάλες ομάδες πληθυσμού χωρίς γνώσεις πληροφορικής μαζί με την αυξανόμενη ανάγκη για κατανάλωση στοχευμένης πληροφορίας δημιουργούν στην απαίτηση παροχής σύντομων, σαφών και περιεκτικών απαντήσεων στα ερωτήματα των χρηστών. Τα συστήματα αυτά αποτελούνται από ένα σύνολο δομικών στοιχείων, για καθένα από τα οποία έχουν προταθεί ποικίλοι τρόποι προσέγγισης από την επιστημονική κοινότητα. Για τη δημιουργία ενός ολοκληρωμένου συστήματος ερωταποκρίσεων πρέπει να μελετηθούν οι υποψήφιες τεχνολογικές προσεγγίσεις, να επιλεγεί η κατάλληλη τεχνική για κάθε δομικό τμήμα του, να γίνει η υλοποίηση σε προγραμματιστικό επίπεδο και τελικά να αναπτυχθεί μία εφαρμογή που θα παρέχει σωστές και σαφείς απαντήσεις στα πλαίσια του σκοπού του οποίου δημιουργήθηκε. Οι διαθέσιμες εργασίες στον τομέα αυτόν για την Ελληνική γλώσσα είναι σαφώς περιορισμένες σε σχέση με την Αγγλική και άλλες γλώσσες.

Σ' αυτή τη μεταπτυχιακή διατριβή αναπτύξαμε ένα σύστημα ερωταποκρίσεων αφού μελετήσαμε τις εναλλακτικές τεχνικές και υλοποιήσαμε ξεχωριστά κάθε δομικό του στοιχείο. Το ολοκληρωμένο σύστημα ερωταποκρίσεων της διατριβής δέχεται κείμενα τα οποία αναλύει και αναπαριστά λεξικολογικά, συντακτικά, λογικά και σημασιολογικά και τα αποθηκεύει με μορφή σημασιολογικών δέντρων. Επιπλέον δέχεται ερωτήσεις τις οποίες επεξεργάζεται με τον ίδιο τρόπο, αναζητάει τα δεδομένα της απάντησης και επιστρέφει στο χρήστη την απάντηση διατυπωμένη σε φυσική γλώσσα. Το σύστημα που αναπτύχθηκε λειτουργεί με κείμενα γραμμένα στην Ελληνική γλώσσα.

1.1 Εισαγωγή

Η αυτόματη συντακτική ανάλυση ενός κειμένου στα πλαίσια της Επεξεργασίας Φυσικής Γλώσσας (Natural Language Processing, NLP) έχει μελετηθεί διεξοδικά και διατίθενται πλέον εμπορικά προϊόντα και υπηρεσίες που αναλύουν κείμενα και παρέχουν πληροφορίες για τη συντακτική δομή μιας πρότασης και των λέξεων που τη συνθέτουν. Υπάρχουν βέβαια ακόμα θέματα προς επίλυση όπως αυτό της αμφισημίας σε λεξικολογικό και συντακτικό επίπεδο. Οι περισσότερες ερευνητικές εργασίες όπως και οι διαθέσιμες εφαρμογές ανάλυσης της φυσικής γλώσσας αφορούν κυρίως στην Αγγλική. Αντίστοιχη έρευνα στα Ελληνικά έχει διεξαχθεί από ελληνικούς κυρίως φορείς και οι σχετικές διαθέσιμες εφαρμογές είναι σαφώς περιορισμένες.

Ένας παρεμφερής τομέας έρευνας που σχετίζεται με την Κατανόηση της Φυσικής Γλώσσας (Natural Language Understanding, NLU) είναι η σημασιολογία (semantics), η οποία αφορά στις έννοιες που εκφράζουν οι λέξεις σε μία φράση και στις μεταξύ τους σχέσεις. Η σημασία κάθε λέξης εξαρτάται από τη θέση της στο σημασιολογικό σύστημα της γλώσσας αλλά και το πλαίσιο των προτάσεων μέσα στο οποίο εμφανίζεται. Η έρευνα στον τομέα της σημασιολογίας έχει προτείνει διάφορους τρόπους αναπαράστασης της γνώσης όπως τα σημασιολογικά δίκτυα (semantic networks) και τα εννοιολογικά δίκτυα (conceptual networks). Με δεδομένη την ύπαρξη επαρκούς γνώσης τόσο στη συντακτική ανάλυση των κειμένων, όσο και στην αναπαράσταση της γνώσης, η πρόκληση βρίσκεται στην τυποποίηση και την εφαρμογή μιας αυτόματης διαδικασίας με την οποία τα κείμενα αναλύονται

συντακτικά και σημασιολογικά και κατασκευάζεται η αναπαράσταση της σημασιολογικής δομής του κειμένου.

Το Μοντέλο Οντοτήτων – Συσχετίσεων (Entity Relationship Model) χρησιμοποιείται για τη μετατροπή απαιτήσεων βάσεων δεδομένων διατυπωμένων σε μορφή κειμένου σε φυσική γλώσσα σε σχήματα βάσεων δεδομένων και δεδομένα σχεσιακών βάσεων. Στην διατριβή χρησιμοποιήσαμε την ιδέα του συγκεκριμένου μοντέλου για να αναπαραστήσουμε κείμενα ως σημασιολογικά δέντρα, δηλαδή να μετατρέψουμε το κείμενο σε αναπαράσταση γνώσης.

Η αναπαράσταση ενός κειμένου ως σημασιολογικό δέντρο παρέχει τη δυνατότητα για προηγμένους τρόπους αναζήτησης, όχι απλά με βάση τη λέξη, αλλά με βάση το νόημα. Ο πιο άμεσος τρόπος για να διεξάγουν οι χρήστες μια τέτοια αναζήτηση είναι να διατυπώνουν τα ερωτήματά τους σε φυσική γλώσσα όπως και να λαμβάνουν τις απαντήσεις σε φυσική γλώσσα.

1.2 Ορισμός του προβλήματος

Στην διατριβή αυτή μελετάμε την αυτόματη αναπαράσταση της σημασιολογικής δομής κειμένων γραμμένων σε φυσική γλώσσα και συγκεκριμένα στην Ελληνική. Τα κείμενα που εισάγονται αναλύονται συντακτικά και στη συνέχεια εφαρμόζονται μια σειρά από κανόνες του μοντέλου οντοτήτων – συσχετίσεων (Entity Relationship Model, ER Model). Το ER Model είναι μια σειρά από οδηγίες για την εξαγωγή δεδομένων και τύπων δεδομένων από κείμενα διατυπωμένα σε φυσική γλώσσα τα οποία χρησιμοποιούνται στην κατασκευή σχημάτων βάσεων δεδομένων (database schemas) και στην καταχώρηση δεδομένων σε σχεσιακές βάσεις δεδομένων. Επιχειρούμε δηλαδή να αξιοποιήσουμε την ιδέα του Μοντέλου Οντοτήτων - Συσχετίσεων για την εξαγωγή από κείμενα των οντοτήτων, των τύπων τους, των ιδιοτήτων και των μεταξύ τους σχέσεων με σκοπό τη δημιουργία μιας αναπαράστασης της σημασιολογικής δομής των κειμένων ως σημασιολογικά δέντρα.

Επιπλέον αξιοποιώντας τη δυνατότητα αναπαράστασης της γνώσης σε ερωτήσεις διατυπωμένες σε φυσική γλώσσα μπορεί να δημιουργηθεί το σημασιολογικό δέντρο που περιέχει το νόημα μιας ερώτησης. Αναζητώντας τη δομή του δέντρου της ερώτησης στο δέντρο αναπαράστασης των κειμένων είναι πιθανό να προκύψει η απάντηση στο ερώτημα ως πληροφορία σε νοηματικό επίπεδο. Η πληροφορία αυτή με τη βοήθεια της γραμματικής της Ελληνικής γλώσσας, μπορεί να δοθεί ως απάντηση στο ερώτημα του χρήστη, σε καλά διατυπωμένη μορφή φυσικής γλώσσας.

Σκοπός της διατριβής είναι η μελέτη και η ανάπτυξη ενός συστήματος ερωταποκρίσεων το οποίο δέχεται κείμενα και ερωτήσεις σε φυσική γλώσσα, τα αναλύει σημασιολογικά και επιχειρεί να παράσχει σωστά διατυπωμένες απαντήσεις στην Ελληνική γλώσσα.

1.3 Η δομή της διατριβής

Η διατριβή είναι δομημένη σύμφωνα με τις παρακάτω ενότητες: Στο Κεφάλαιο 2 παρουσιάζεται το θεωρητικό υπόβαθρο και η σχετική δουλειά στον τομέα της αυτόματης μετατροπής και αναπαράστασης γνώσης από κείμενα γραμμένα σε φυσική γλώσσα όπως επίσης και η δουλειά σχετικά με τα συστήματα ερωταποκρίσεων. Στο Κεφάλαιο 3 παρουσιάζεται η επισκόπηση του συστήματος ερωταποκρίσεων που αναπτύξαμε και τα χαρακτηριστικά του ενώ στο Κεφάλαιο 4 περιγράφεται η γενική αρχιτεκτονική του συστήματός μας. Στο κεφάλαιο 5 περιγράφονται αναλυτικά όλα τα δομικά

μέρη του συστήματος με παραδείγματα. Στο κεφάλαιο 6 παρουσιάζονται παραδείγματα από τη χρήση του και την εμπειρία που παρέχει στον τελικό χρήστη. Στο κεφάλαιο 7 προτείνονται μελλοντικές επεκτάσεις του συστήματος και στο κεφάλαιο 8 συζητούνται τα συμπεράσματα από την ανάπτυξή του.

2. Θεωρητικό υπόβαθρο και σχετικές εργασίες

2.1 Επεξεργασία φυσικής γλώσσας

Η Επεξεργασία Φυσικής Γλώσσας (Natural Language Processing) ανήκει στον τομέα της Τεχνητής Νοημοσύνης και έχει σκοπό την κατανόηση αλλά και τη σύνθεση εκφράσεων σε ανθρώπινη γλώσσα [34]. Η προφανής χρήση της είναι στην επικοινωνία ανθρώπου – μηχανής. Με την εξέλιξη των τεχνολογιών η «μηχανή» δεν θεωρείται πλέον μόνο ο υπολογιστής αλλά μπορεί να είναι το tablet, το κινητό τηλέφωνο, το αυτοκίνητο, οι ηλεκτρικές συσκευές κλπ. Η Επεξεργασία Φυσικής Γλώσσας βρίσκει επίσης εφαρμογή στη διαχείριση μη δομημένων πληροφοριών διατυπωμένων σε φυσική γλώσσα που είναι αποθηκευμένες σε ηλεκτρονική μορφή δηλαδή έναν τεράστιο όγκο δεδομένων που βρίσκεται σε ιστοσελίδες, κοινωνικά δίκτυα και αρχεία. Η διαχείριση μπορεί να ξεκινά από την κατηγοριοποίηση του εγγράφου και να φτάνει έως την απάντηση ερωτήσεων που αφορούν σε πληροφορίες που αναφέρονται σε αυτό. Άλλη εφαρμογή είναι η υποβολή ερωτημάτων ως κείμενο για αναζήτηση πληροφοριών σε βάσεις δεδομένων από απλούς χρήστες χωρίς εξειδικευμένες γνώσεις. Κοινός παράγοντας όλων αυτών είναι η εξαγωγή του νοήματος από την πληροφορία σε αντίθεση με τη διαχείριση των κειμένων σε επίπεδο χαρακτήρων που πραγματοποιείται σε μια απλή αναζήτηση. Γι αυτό σε ορισμένες περιπτώσεις γίνεται λόγος για Κατανόηση Φυσικής Γλώσσας (Natural Language Understanding) [20].

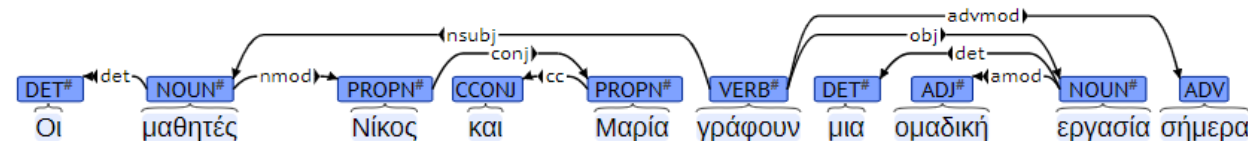
Ένα βασικό πρόβλημα στην αυτόματη ανάλυση της φυσικής γλώσσας είναι η ασάφεια (ambiguity) η οποία δημιουργεί διφορούμενη ερμηνεία. Η ασάφεια μπορεί να είναι σε επίπεδο συντακτικό, λεξικολογικό, αναφορικό (σε ποιον/τι), σημασιολογικό και πραγματολογικό.

Η Επεξεργασία της Φυσικής Γλώσσας περιλαμβάνει το στάδιο της συντακτικής ανάλυσης, της σημασιολογικής ανάλυσης και της πραγματολογικής ανάλυσης. Η συντακτική ανάλυση καθορίζει τον τρόπο με τον οποίο διατάσσονται οι ομάδες από σύμβολα ώστε να θεωρηθεί ότι σχηματίζουν μια καλοσχηματισμένη (well-formed) πρόταση. Η σημασιολογική ανάλυση καθορίζει ποιο θεωρείται το νόημα των καλοσχηματισμένων προτάσεων. Η πραγματολογική ανάλυση καθορίζει τον τρόπο με τον οποίο πληροφορίες από τα συμφραζόμενα μπορούν να αξιοποιηθούν για την αποσαφήνιση των σχέσεων μεταξύ διαφορετικών σημασιολογιών [11]. Τα στάδια της ανάλυσης δεν είναι απαραίτητα διακριτά και σειριακά.

Οι λέξεις αποθηκεύονται στη βασική τους μορφή στο λεξικό. Οι υπόλοιπες μορφές προκύπτουν από τη μορφολογία των κλίσεων, προσθήκη προθεμάτων και καταλήξεων και τη σύνθεση νέων λέξεων. Η γραμματική ανάλυση των προτάσεων γίνεται με βάση τη γραμματική που προσφέρει η δηλωτική αναπαράσταση των συντακτικών στοιχείων μιας γλώσσας και ενός συντακτικού αναλυτή ο οποίος συγκρίνει την πρόταση που εισάγεται με τη γραμματική και προσπαθεί να ταιριάξει τη μορφή της πρότασης με τους συντακτικούς κανόνες της γλώσσας.

Με τη Γραμματική Οριστικών Προτάσεων (definite clause grammar, DCG) μπορεί να εκφραστεί η γραμματική μιας φυσικής ή τυπικής γλώσσας. Η Γραμματική Οριστικών Προτάσεων αποτελείται από τρία συντακτικά μέρη: α) Τα τερματικά σύμβολα (terminals) τα οποία είναι οι λέξεις που διαμορφώνουν το λεξικό. β) Τα μη τερματικά σύμβολα (non-terminals) τα οποία περιγράφουν τις δομές της γλώσσας δηλαδή τις λεκτικές κατηγορίες (π.χ. ουσιαστικά, ρήματα), τις συντακτικές κατηγορίες (π.χ. υποκείμενο, κατηγορηματική πρόταση) και το αρχικό σύμβολο το οποίο αντιπροσωπεύει την αρχική πρόταση που περιγράφεται γραμματικά. γ) Τους κανόνες παραγωγής (production rules ή syntax rules) που περιγράφουν τους έγκυρους συνδυασμούς των κανόνων της γλώσσας (π.χ Π → ΟΦ ΡΦ όπου Π: το αρχικό σύμβολο, ΟΦ η ονοματική φράση και ΡΦ η ρηματική φράση).

Στο Σχήμα 1 φαίνεται η συντακτική ανάλυση της φράσης «Οι μαθητές Νίκος και Μαρία γράφουν μια ομαδική εργασία σήμερα». Εμφανίζεται το υποκείμενο, το ρήμα και το αντικείμενο, ο τρόπος που συνδέονται οι λέξεις καθώς και ο χαρακτηρισμός κάθε λέξης της πρότασης. Η ανάλυση έγινε από το online εργαλείο του Ινστιτούτου Επεξεργασίας Λόγου [25] . Οι ετικέτες που χρησιμοποιούνται προέρχονται από το Universal Dependencies [49] [17] .



Σύμβολα:

DET	Determiner	Προσδιοριστής
NOUN	Noun	Ουσιαστικό
PROP	Proper Noun	Κύριο όνομα
CCONJ	Coordinating conjunction	Συμπλεκτικός σύνδεσμος
VERB	Verb	Ρήμα
ADJ	Adjective	Επίθετο
ADV	Adverb	Επίρρημα
nsubj	Nominal subject	Υποκείμενο
conj	Conjunct	Συνένωση
advmod	Adverbial modifier	Επιρρηματικός Προσδιορισμός
obj	Object	Αντικείμενο
amod	adjectival modifier	Επιθετικός Τροποποιητής

Σχήμα 1. Η συντακτική ανάλυση της πρότασης «Η γραμματική οριστικών προτάσεων αποτελείται από τρία συντακτικά μέρη» από το service του ILSP NLP tools [25]

Ο συντακτικός αναλυτής (parser) δημιουργεί το συντακτικό δέντρο της πρότασης (parse tree) χρησιμοποιώντας τη γραμματική της γλώσσας. Η ρίζα του δένδρου είναι το αρχικό σύμβολο της γραμματικής δηλαδή η πρόταση, οι κόμβοι αντιπροσωπεύουν τα μη τερματικά σύμβολα και τα φύλλα του αντιπροσωπεύουν τα τερματικά σύμβολα. Η δομή του αποτελείται από τα συντακτικά μέρη της πρότασης τοποθετημένα ιεραρχικά. Ο συντακτικός αναλυτής αναπτύσσει το συντακτικό δέντρο εφαρμόζοντας τους κανόνες γραμματικής μέχρι τα φύλλα δηλαδή τα τερματικά σύμβολα. Στο Σχήμα 2 φαίνεται το δέντρο συντακτικής ανάλυσης (Parse Tree) της πρότασης «Ο σκύλος κυνηγάει τη γάτα».



Σχήμα 2. Το δέντρο συντακτικής ανάλυσης (Parse Tree) της πρότασης «Ο σκύλος κυνηγάει τη γάτα»

Η σημασιολογική ανάλυση πραγματοποιείται με την μετατροπή των προτάσεων σε εσωτερικές δομές αναπαράστασης γνώσης χρησιμοποιώντας τη νοηματική σημασία των λέξεων. Για την πραγματολογική ανάλυση το σύστημα πρέπει να διαθέτει γνώση του γενικότερου νοηματικού πλαισίου στο οποίο αναφέρεται η πρόταση ώστε να βγάλει εύλογα συμπεράσματα [20] .

Για τη σύνθεση του κειμένου το οποίο απευθύνεται προς το χρήστη του συστήματος και αποτελεί την απάντηση στο ερώτημά του, εφαρμόζονται στην πληροφορία οι κανόνες γραμματικής και παράγεται μια σωστά διατυπωμένη πρόταση σε φυσική γλώσσα.

2.2 Αναπαράσταση Γνώσης

Η Γνώση είναι πληροφορίες των οποίων έχει ελεγχθεί η ορθότητα. Η γνώση διακρίνεται σε δηλωτική και διαδικαστική. Η Δηλωτική Γνώση (*declarative knowledge*) είναι γνώση για τα αντικείμενα (*objects-data*), τα γεγονότα (*facts*) και τα συμβάντα (*events*). Με τη δηλωτική γνώση περιγράφεται μία κατάσταση ή ένα πρόβλημα προς επίλυση. Η Διαδικαστική Γνώση (*procedural knowledge*) είναι η γνώση για τη διεκπεραίωση μιας διαδικασίας. Με τη διαδικαστική γνώση περιγράφεται ο τρόπος λύσης ενός προβλήματος. Η δηλωτική γνώση χωρίζεται στη Σημασιολογική Γνώση και την Επεισοδιακή Γνώση. Η Σημασιολογική Γνώση (*semantic knowledge*) είναι η γνώση για έννοιες που αντιπροσωπεύουν τα αντικείμενα (*οντότητες*) του κόσμου και τις μεταξύ τους σχέσεις. Η Επεισοδιακή Γνώση είναι γνώση για εμπειρίες και γεγονότα του παρελθόντος και είναι οργανωμένη σε σειριακή μορφή [20] .

Ο πιο ευρέως χρησιμοποιούμενος τρόπος αναπαράστασης της γνώσης είναι η Φυσική Γλώσσα. Η Φυσική Γλώσσα όμως δε μπορεί να επεξεργαστεί από υπολογιστικά συστήματα τα οποία απαιτούν συγκεκριμένο συντακτικό και σημασιολογία. Οι έννοιες της φυσικής γλώσσας πρέπει να μεταφερθούν με συγκεκριμένους κανόνες σε πιο «αυστηρές» συντακτικά δομές οι οποίες γίνονται εύκολα επεξεργάσιμες. Για την αναπαράσταση της γνώσης χρησιμοποιείται μονοσήμαντη και τυποποιημένη σύνταξη, χωρίς ασάφειες και χωρίς συμφραζόμενα και δυνατότητα να συνδυαστεί με μηχανισμούς εξαγωγής συμπερασμάτων. Οι σημαντικότερες μέθοδοι αναπαράστασης γνώσης στα πλαίσια της

τεχνητής νοημοσύνης είναι η Λογική, οι Δομημένες Μορφές Αναπαράστασης Γνώσης, οι Κανόνες και οι Οντολογίες.

Η λογική περιλαμβάνει τον Προτασιακό Λογισμό, τον Κατηγορηματικό Λογισμό ή Κατηγορηματικό Λογισμό Πρώτης Τάξης. Ο Προτασιακός Λογισμός (*propositional calculus*) βασίζεται σε δηλωτικές προτάσεις που μπορεί να είναι αληθείς ή ψευδείς. Ο Κατηγορηματικός Λογισμός (*predicate calculus*) επεκτείνει τον Προτασιακό ώστε να καθορίζονται σχέσεις μεταξύ αντικειμένων του πραγματικού κόσμου. Για την αναπαράσταση των γνώσεων χρησιμοποιούνται σύμβολα (*symbols*), όπως οι ποσοδείκτες, συναρτησιακά σύμβολα (*functional symbols*) και κατηγορήματα (*predicates*). Ο Κατηγορηματικός Λογισμός επιτρέπει την εξαγωγή συμπερασμάτων με λογικούς συνειρμούς.

Οι Δομημένες Μορφές Αναπαράστασης Γνώσης περιλαμβάνουν τα Σημασιολογικά Δίκτυα, τα Σχήματα, τα Πλαίσια και τα Σενάρια. Τα Σημασιολογικά Δίκτυα αναπαριστούν σημασιολογικές σχέσεις μεταξύ ιδεών σε κάποιον συγκεκριμένο τομέα. Οι ιδιότητες κληρονομούνται και είναι δυνατή η εξαγωγή συμπερασμάτων. Η αναπαράσταση γίνεται με κόμβους, συνδέσεις και χαρακτηρισμούς συνδέσεων. Κάθε σύνδεση συνδέει ένα κόμβο-ουρά με έναν κόμβο-κεφαλή [20].

Τα Σχήματα (*schemata*) είναι δομημένες κλάσεις εννοιών που περιέχουν γενική γνώση η οποία χρησιμοποιείται για αναπαράσταση αντικειμένων, σχέσεων, καταστάσεων, γεγονότων ή αλληλουχίας γεγονότων. Τα Πλαίσια (*frames*) αποτυπώνουν δηλωτική γνώση και περιγράφουν μία ανώτερη κλάση (*class*) ή ένα συγκεκριμένο στοιχείο (*instance frames*). Τα Σενάρια (*scripts*) αποτελούν μία συγκεκριμένη κατηγορία Σχημάτων για την αναπαράσταση επεισοδιακής γνώσης με την παράθεση Πλαισίων σε χρονική σειρά.

Οι Κανόνες (*rules*) αποτελούν μία μέθοδο αναπαράστασης διαδικαστικής γνώσης που αντιστοιχεί σε έγκυρους συλλογισμούς και έχει τη μορφή AN-TOTE.

Η Οντολογία (*ontology*) είναι μια τυπική αναπαράσταση γνώσης ως ένα σύνολο εννοιών, σχέσεων και ιδιοτήτων που μπορεί να χρησιμοποιηθεί για συλλογιστική (εξαγωγή συμπερασμάτων και νέας γνώσης) και για την δομημένη περιγραφή γνώσης ενός πεδίου ενδιαφέροντος [56].

Στον Πίνακα 1 φαίνονται οι δημοφιλέστερες σχολές στην αναπαράσταση γνώσης και οι αντίστοιχες αναφορές στον τομέα της φυσικής γλώσσας [11].

APPROACH	CHARACTERISTIC FEATURES	REFERENCE
PRODUCTION RULE	CYCLES OF 'RECOGNIZE', 'RESOLVE CONFLICT', 'ACT' STEPS	(Chomsky, 1956) [15]
SEMANTIC PATTERN MATCHING	SEMANTIC CATEGORIES AND SEMANTIC CASE FRAMES	(Ceccato, 1967) [12]
FIRST ORDER LOGIC (FOL)	AXIOMS AND RULES OF INFERENCES	(Barwise, 1977) [9]
BAYESIAN NETWORKS	VARIABLES REPRESENTED BY A PROBABILISTIC DIRECTED ACYCLIC GRAPH	(Pearl, 1985) [41]
SEMANTIC NETWORKS	PATTERNS OF INTERCONNECTED NODES AND ARCS	(Sowa, 1987) [47]
ONTOLOGY WEB LANGUAGE (OWL)	HIERARCHICAL CLASSES AND RELATIONSHIPS BETWEEN THEM	(McGuinness, 2004) [32]

Πίνακας 1. Οι δημοφιλέστερες σχολές στην αναπαράσταση γνώσης και οι αντίστοιχες αναφορές στον τομέα της φυσικής γλώσσας [11]

2.3 Συστήματα Ερωταποκρίσεων (Question Answering Systems)

Στις μέρες μας υπάρχει τεράστιος όγκος πληροφοριών στο web στον οποίο οι χρήστες πραγματοποιούν αναζητήσεις μέσα από τις μηχανές αναζήτησης, λαμβάνοντας ως απάντηση λίστες από συνδέσμους σε ιστοσελίδες. Αν και αυτός ο καθιερωμένος τρόπος αναζήτησης στις περισσότερες περιπτώσεις δίνει ικανοποιητικά αποτελέσματα, οι χρήστες που έχουν συγκεκριμένα ερωτήματα, θα ήθελαν να τα υποβάλουν σε φυσική γλώσσα και να λαμβάνουν μία άμεση απάντηση διατυπωμένη με ακρίβεια.

Τα συστήματα ερωταποκρίσεων επιχειρούν να δώσουν ακριβείς απαντήσεις σε ερωτήσεις διατυπωμένες σε φυσική γλώσσα χρησιμοποιώντας δεδομένα από ηλεκτρονικά έγγραφα ή από βάσεις δεδομένων [5]. Εισήχθησαν περίπου το 1970 και ήταν περιορισμένα σε συγκεκριμένους τομείς. Σήμερα σημαντικά πεδία εφαρμογής τέτοιων συστημάτων αποτελούν η ανάκτηση πληροφοριών από το web μέσω έξυπνων μηχανών αναζήτησης, από δημόσιες και ιδιωτικές βάσεις δεδομένων και από αιτήματα σε μεμονωμένες ιστοσελίδες [5]. Επιπλέον η χρήση τους επεκτείνεται σε πηγές πολυμέσων με αναζήτηση αρχείων video, φωτογραφιών και ήχου. Ένας παράγοντας που εντείνει την ανάγκη για τεχνολογίες ερωταποκρίσεων είναι η αυξανόμενη χρήση και τα εξελιγμένα χαρακτηριστικά των κινητών συσκευών. Η δυνατότητά τους να δέχονται προφορικές ερωτήσεις από τους χρήστες και να τους προσφέρουν φωνητικές απαντήσεις αποτελεί το σύγχρονο τρόπο αλληλεπίδρασης σε φυσική γλώσσα που αυξάνει την ευχρηστία και διευρύνει την αποδοχή των συστημάτων ερωταποκρίσεων από μεγάλες ομάδες χρηστών.

Συστήματα ερωταποκρίσεων υπάρχουν εμπορικά διαθέσιμα όπως το Siri, ο προσωπικός βοηθός (personal assistant) των προϊόντων της Apple και «εξερευνητής γνώσης» [24] [46], ο προσωπικός βοηθός της Google που είναι ενσωματωμένος στο λειτουργικό σύστημα Android [22], το υπολογιστικό σύστημα ερωταποκρίσεων της IBM με το όνομα Watson που σχεδιάστηκε ειδικά για να απαντάει ερωτήσεις τύπου κουίζ στην εκπομπή Jeopardy [51] και το Wolfram|Alpha το οποίο απαντάει σε

ερωτήσεις εκτελώντας υπολογισμούς χρησιμοποιώντας μια μεγάλη συλλογή από ενσωματωμένα δεδομένα, αλγόριθμους και μεθόδους [52] .

Η ανάπτυξη των συστημάτων ερωταποκρίσεων προϋποθέτει γνώσεις από τον τομέα των τεχνολογιών πληροφορικής, τεχνητής νοημοσύνης, επεξεργασίας φυσικής γλώσσας, διαχείρισης γνώσης, διαχείρισης βάσεων δεδομένων και γνωσιακή επιστήμη. Για την υλοποίησή τους χρησιμοποιείται η επεξεργασία φυσικής γλώσσας, η ανάκτηση πληροφοριών, η αναπαράσταση γνώσης και η συλλογιστική [43] .

Ως *ερώτημα* ορίζεται μια πρόταση σε φυσική γλώσσα, η οποία συνήθως αρχίζει με μία ερωτηματική λέξη και εκφράζει μια ανάγκη του χρήστη για πληροφορία. Οι ερωτήσεις σε φυσική γλώσσα ενώ αποτελούν πολύ καλά καθορισμένα αιτήματα για πληροφορία, παράλληλα μεταφέρουν περισσότερη πληροφορία από μια απλή λίστα όρων αναζήτησης καθώς αναπαριστούν συντακτικές και σημασιολογικές σχέσεις μεταξύ των όρων αναζήτησης [26] .

Ως τύπος ερωτήματος ορίζεται μια συγκεκριμένη σημασιολογική κατηγορία ερωτήσεων οι οποίες χαρακτηρίζονται από κοινές ιδιότητες. Οι κύριοι τύποι ερωτήσεων είναι αυτές οι οποίες δέχονται ως απάντηση *γεγονότα, λίστες, ορισμούς, υποθέσεις, αιτιολογήσεις, σχέσεις, διαδικασίες και επιβεβαίωση*. Οι ερωτήσεις σχετικές με *γεγονότα* επιδέχονται σύντομες απαντήσεις βασισμένες σε κάποιο γεγονός όπως ονόματα και ημερομηνίες και ξεκινούν συνήθως με τα ερωτηματικά Τι, Πότε, Που και Ποιος. Για παράδειγμα «Πότε γεννήθηκε ο Χ» ή «Ποια χρονιά συνέβη το Υ». Η σύγχρονη έρευνα στρέφεται σε πιο σύνθετους τύπους ερωτήσεων οι οποίες χρειάζονται περιγραφικές απαντήσεις. Τέτοιους είδους ερωτήσεις είναι για παράδειγμα ερωτήσεις βιογραφίας όπως «Ποιος είναι ο Αλβέρτος Αϊνστάιν;», ερωτήσεις *ορισμού οντοτήτων* όπως «Τι είναι το DNA;», ερωτήσεις δημιουργίας *λίστας* όπως «Ποιες χώρες ανήκουν στην Ευρωπαϊκή Ένωση;», ερωτήσεις που βασίζονται σε *σενάριο* (και ζητούν σχέσεις ανάμεσα σε οντότητες του σεναρίου) και ερωτήσεις *αιτιολόγησης* τύπου «Γιατί...;» [50] .

Οι διαφορετικοί τύποι των συστημάτων ερωταποκρίσεων, χωρίζονται σε δύο κύριες κατηγορίες, ανάλογα με τη μέθοδο που χρησιμοποιούν. Η πρώτη κατηγορία συστημάτων είναι αυτά που βασίζονται στην *ανάκτηση πληροφοριών για γεγονότα (Information Retrieval based factoid)* και χρησιμοποιούν απλές μεθόδους επεξεργασίας φυσικής γλώσσας και ανάκτησης δεδομένων. Η δεύτερη κατηγορία βασίζεται στη *γνώση και τη συλλογιστική* [43] . Η σύγκριση των δύο κατηγοριών φαίνεται στον Πίνακα 2. Η μηχανή αναζήτησης της Google είναι ένα παράδειγμα της πρώτης κατηγορίας, το Siri της Apple είναι ένα παράδειγμα της δεύτερης και το Watson της IBM είναι ένα παράδειγμα υβριδικής προσέγγισης.

	Συστήματα Ερωταποκρίσεων βασισμένα σε Επεξεργασία Φυσικής Γλώσσας και Ανάκτηση Πληροφοριών (QA system based on NLP and IR)	Συστήματα Ερωταποκρίσεων βασισμένα σε Συλλογιστική με Επεξεργασία Φυσικής Γλώσσας (QA systems Reasoningwith NLP)
Τεχνική	Συντακτική ανάλυση, Εξαγωγή οντοτήτων και Ανάκτηση Πληροφοριών (Information Retrieval)	Σημασιολογική Ανάλυση ή Συλλογιστική
Προέλευση εγγράφων	Έγγραφα κειμένου	Βάση γνώσης
Πεδίο	Ανοιχτό Πεδίο	Περιορισμένο Πεδίο
Απόκριση	Εξαγόμενα αποσπάσματα κειμένου	Σύνθεση απόκρισης
Τύπος ερωτήσεων	Ποιος/Πότε/Τι/ (wh- type of questions)	Περισσότερα είδη ερωτήσεων

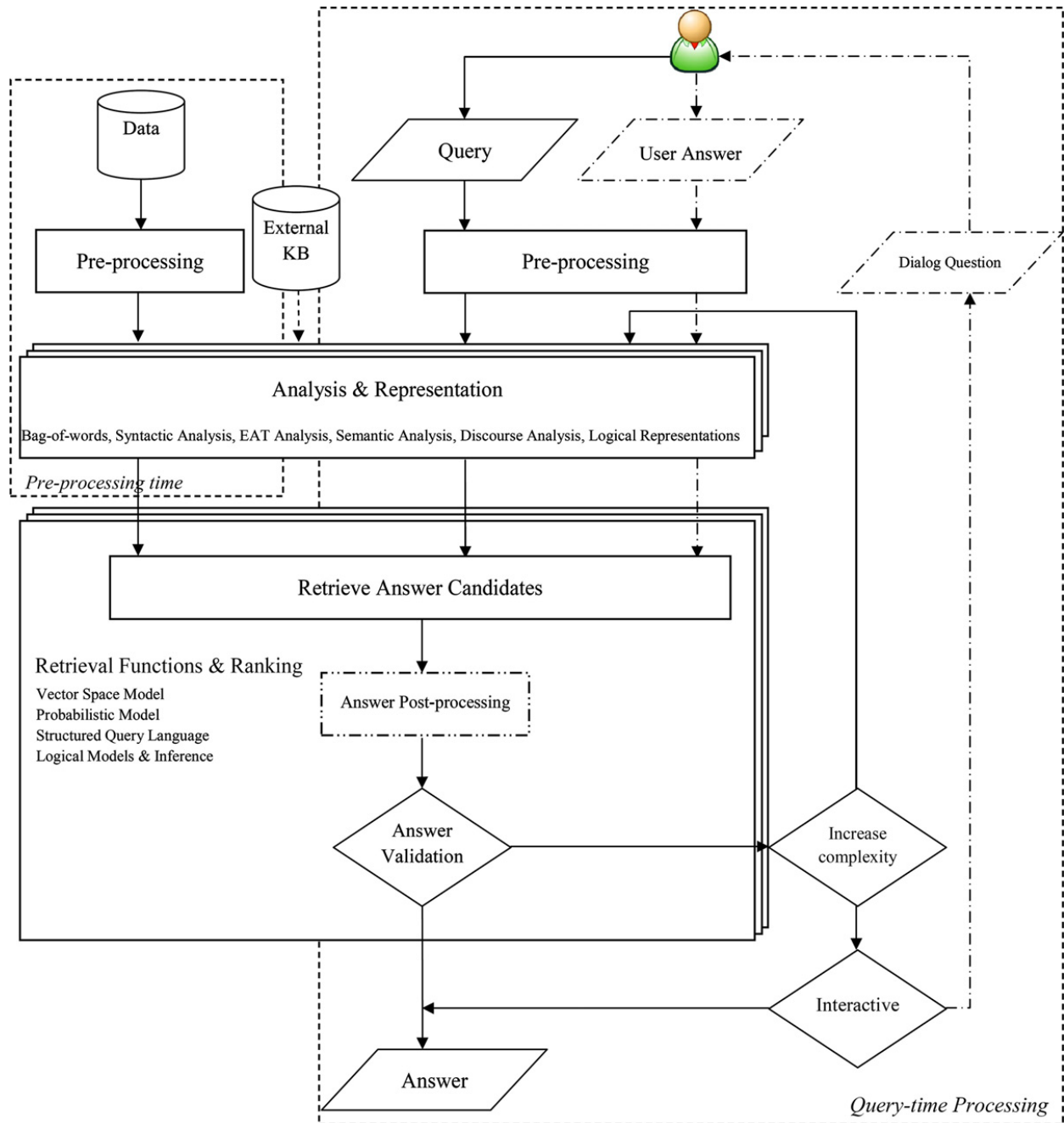
Πίνακας 2. Κατηγοριοποίηση των συστημάτων ερωταποκρίσεων [43].

Τα συστήματα ερωταποκρίσεων ταξινομούνται επίσης σε δύο κατηγορίες ανάλογα με το αν είναι ή όχι περιορισμένα σε ένα πεδίο (domain). Τα συστήματα που μπορούν να αποκριθούν σε οτιδήποτε μπορεί να δεχτεί απάντηση ονομάζονται συστήματα ερωταποκρίσεων ανοιχτού πεδίου (open-domain), ενώ αντίθετα τα συστήματα που δέχονται ερωτήσεις μόνο πάνω σε συγκεκριμένο πεδίο ονομάζονται συστήματα ερωταποκρίσεων κλειστού πεδίου ή περιορισμένου πεδίου (closed-domain ή restricted-domain) όπως για παράδειγμα ιατρική, μουσική, καιρός [33]. Συνήθως τα συστήματα κλειστού πεδίου απευθύνονται σε χρήστες που γνωρίζουν την ορολογία του πεδίου τους, όπως π.χ. ιατρική και βασίζονται σε δεδομένα που έχουν κατασκευαστεί χειροκίνητα [26].

Στα συστήματα που βασίζονται σε ανάκτηση πληροφορίας (IR-based systems) οι ερωτήσεις που εισάγονται από τους χρήστες επεξεργάζονται για την εξαγωγή του τύπου της ερώτησης και το σχηματισμό του ερωτήματος (query formulation). Στη συνέχεια γίνεται η ανάκτηση των εγγράφων και λαμβάνονται τα αποσπάσματα εκείνα τα οποία είναι σχετικά με την ερώτηση και που αποτελούν υποψήφια απαντήσεις. Τα αποσπάσματα κατατάσσονται με βάση στοιχεία που λαμβάνονται από το κείμενο ή άλλα κριτήρια ανάλογα με τον αλγόριθμο κατάταξης (ranking). Τα συστήματα αυτά βασίζονται σε απλοποιημένη αναπαράσταση των ερωτημάτων και απλές και γρήγορες τεχνικές ανάκτησης πληροφοριών χωρίς να εκτελούν σε βάθος ανάλυση, ούτε των ερωτήσεων, ούτε των εγγράφων. Δεν επιστρέφουν απάντηση στο χρήστη, αλλά μια σειρά ταξινομημένων εγγράφων τα οποία περιέχουν κείμενα σχετικά με την ερώτηση.

Αντίθετα τα συστήματα επεξεργασίας φυσικής γλώσσας μετατρέπουν το κείμενο σε αυστηρά καθορισμένη αναπαράσταση του νοήματος όπως κατηγορηματική λογική πρώτης τάξης (first order predicate calculus), σημασιολογικά δίκτυα (semantic networks), εννοιολογικούς χάρτες (conceptual dependency diagrams) και αναπαραστάσεις βασισμένες σε πλαίσια (frame-based representations) [5]. Όταν εντοπιστεί η υποψήφια απάντηση σε ένα έγγραφο από το σύστημα επεξεργασίας φυσικής

γλώσσας εξάγεται μόνο η σχετική λέξη ή φράση από το κείμενο η οποία απαντάει στην ερώτηση. Η τεχνολογία εξαγωγής της απάντησης αποτελεί καθοριστικό παράγοντα για τα αποτελέσματα ενός συστήματος ερωταποκρίσεων [2]. Για την επίτευξη των σωστών απαντήσεων σε δεδομένες ερωτήσεις, δεν απαιτείται μόνο η δημιουργία μιας συλλογής σωστών απαντήσεων, αλλά και η κατανόηση των μεταξύ τους σχέσεων [33].



Σχήμα 3: Η αρχιτεκτονική ενός συστήματος ερωταποκρίσεων σε υψηλό επίπεδο (Kolomiyets, 2011) [26]

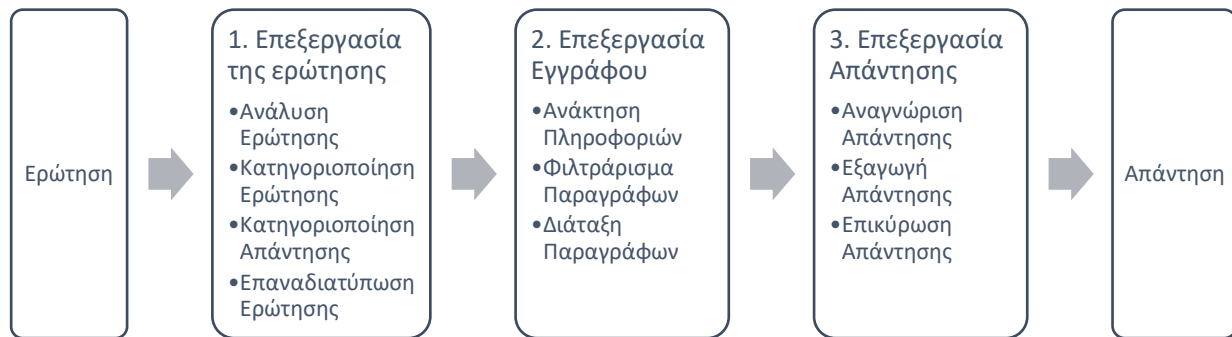
Στο Σχήμα 3 φαίνεται η γενική αρχιτεκτονική ενός συστήματος ερωταποκρίσεων όπως περιγράφεται στη δημοσίευση “A survey on question answering technology from an information retrieval perspective” [26]. Οι ερωτήσεις σε φυσική γλώσσα, καθώς και το περιεχόμενο των εγγράφων επεξεργάζονται και μεταφράζονται σε μορφές που είναι αναγνωρίσιμες από τους υπολογιστές (Analysis & Representation). Η απάντηση συνήθως προκύπτει από τη συμπλήρωση μιας κενής θέσης στο πλαίσιο της αντιστοίχισης της αναπαράστασης της ερώτησης με την αναπαράσταση της πρότασης. Σε περίπτωση διαφορετικών πιθανών απαντήσεων ή απάντησης που δεν ικανοποιεί πλήρως το ερώτημα, εφαρμόζεται στις πιθανές απαντήσεις μια σειρά κατάταξης (ranking) σύμφωνα με τη σχετικότητά τους με τη ζητούμενη πληροφορία. Σε ορισμένες περιπτώσεις οι απαντήσεις χρειάζονται διαμόρφωση (answer post processing) πριν εμφανιστούν στον χρήστη. Στο ίδιο σχήμα φαίνεται επίσης η δυνατότητα να ζητηθούν περισσότερες πληροφορίες από το χρήστη ώστε να συμπληρωθούν τα δεδομένα της ερώτησης σε περίπτωση που δε βρεθεί απάντηση ή αυτή δεν κριθεί ικανοποιητική.

Οι μέθοδοι που χρησιμοποιούνται στα συστήματα ερωταποκρίσεων ξεκινώντας από τις πιο απλές είναι [26]:

1. **Bag of words:** Αφαιρούνται από την πρόταση οι συνηθισμένες λέξεις που δεν της δίνουν σημασία και συγκρίνονται οι υπόλοιπες, χωρίς να διατηρηθούν δομικά χαρακτηριστικά της πρότασης.
2. **Μορφολογική και συντακτική ανάλυση (Morpho-syntactic analysis):** Οι ερωτήσεις και οι προτάσεις αναπαριστάνονται με μια συντακτική δομή στην οποία κόμβοι αποτελούν οι λέξεις που δίνουν σημασία στην πρόταση (ουσιαστικά, ρήματα, επίθετα και επιρρήματα) οι οποίες ενώνονται μεταξύ τους δημιουργώντας το δέντρο εξαρτήσεων. Τα δέντρα εξαρτήσεων των ερωτήσεων και των προτάσεων συγκρίνονται για την εξαγωγή των πιθανών απαντήσεων.
3. **Σημασιολογική κατηγοριοποίηση της αναμενόμενης απάντησης:** Από την ανάλυση της ερώτησης προκύπτει η σημασιολογική κλάση της υποψήφιας απάντησης.
4. **Σημασιολογική κατηγοριοποίηση όλων των λέξεων:** Οι ερωτήσεις και οι προτάσεις αναπαριστάνονται ως πλαίσια (frames) ή δηλωτικές δομές που περιγράφουν τους σημασιολογικούς ρόλους. Τα ερωτήματα μπορούν να αναπαρασταθούν ως τριπλέτες και να αποσαφηνιστούν βασικοί συντακτικοί ρόλοι όπως το υποκείμενο και το αντικείμενο.
5. **Μετατροπή και ανάκτηση με δομημένη γλώσσα ερωτημάτων:** Οι ερωτήσεις μετατρέπονται σε δομημένη μορφή η οποία είναι εύκολο να εκφραστεί ως ερώτημα sql (structured query language) στη βάση δεδομένων. Σε αυτή την περίπτωση τα δεδομένα ανακτώνται μόνο εφόσον ικανοποιηθούν οι συνθήκες του ερωτήματος. Δεν υπάρχει τρόπος διαχείρισης αβεβαιότητας ούτε απαντήσεις διαφορετικής βαρύτητας.
6. **Μετατροπή σε λογική αναπαράσταση και χρήση συλλογιστικής:** Οι ερωτήσεις και οι προτάσεις αναπαρίστανται με λογική πρώτης τάξης (first-order logic) ή άλλες λογικές αναπαραστάσεις. Η μετατροπή αυτή έχει τη δυνατότητα να συμπεριλάβει σημασιολογική πληροφορία. Η εξαγωγή των υποψήφιας απαντήσεων γίνεται με λογικές πράξεις και συλλογιστική (reasoning).

Τα περισσότερα συστήματα ερωταποκρίσεων αποτελούνται από τρία τμήματα: *Τον επεξεργαστή της ερώτησης, τον επεξεργαστή του εγγράφου και τον επεξεργαστή της απάντησης.* Όταν στο σύστημα ερωταποκρίσεων εισαχθεί μία ερώτηση, το πρώτο βήμα στην επεξεργασία της είναι να εντοπιστεί ο τύπος της ερώτησης και ακολούθως να καθοριστεί ο τύπος της απάντησης. Στις περιπτώσεις που

χρειάζεται, η ερώτηση επαναδιατυπώνεται και στη συνέχεια εισάγεται στο σύστημα ανάκτησης πληροφοριών ώστε να γίνει αναζήτηση των εγγράφων βασισμένη σε λέξεις κλειδιά. Από τα έγγραφα που επιστρέφονται απομονώνονται τα αποσπάσματα που σχετίζονται με την απάντηση και εκτελείται η κατάταξη των εγγράφων σύμφωνα με τη σχετικότητα τους με το ερώτημα. Από τις λέξεις ή φράσεις που απομονώνονται δημιουργείται η απάντηση η οποία ελέγχεται για την ορθότητά της πριν εμφανιστεί στο χρήστη. Στις περιπτώσεις που τα δεδομένα είναι αποθηκευμένα σε δομημένη μορφή όπως π.χ. σε βάση δεδομένων, σχηματίζεται και υποβάλλεται το ερώτημα προς της βάση με τα αποτελέσματα του οποίου δημιουργείται η τελική απάντηση. Στο *Σχήμα 4* φαίνονται τα τμήματα ενός συστήματος ερωταποκρίσεων που πραγματοποιεί αναζήτηση σε έγγραφα.



Σχήμα 4: Η Δομή ενός συστήματος ερωταποκρίσεων (Allam, 2012) [2]

Η εξαγωγή πληροφορίας για γεγονότα από ψηφιακές πηγές και η δημιουργία απάντησης είναι ένας τομέας στον οποίο έχει συντελεστεί σημαντική πρόοδος. Υπάρχει όμως πεδίο για έρευνα στον τομέα της εξαγωγής εννοιολογικής πληροφορίας καθώς και στον τομέα της σύνδεσης συναφών πληροφοριών που βρίσκονται σε ένα ή σε διαφορετικά έγγραφα.

2.4 Μοντέλο Οντοτήτων – Συσχετίσεων (ER Model)

Το μοντέλο Οντοτήτων – Συσχετίσεων (ER Model) είναι ένα εννοιολογικό μοντέλο υψηλού επιπέδου το οποίο περιγράφει την πληροφορία ως σύνολο από οντότητες (entities), ιδιότητες (attributes) και σχέσεις (relationships) [10]. Χρησιμοποιείται για την ανάλυση των απαιτήσεων ενός πληροφοριακού συστήματος και περιγράφει τις πληροφορίες οι οποίες αποθηκεύονται σε βάσεις δεδομένων και τους τύπους τους [55]. Η ιδέα του μοντέλου Οντοτήτων – Συσχετίσεων προτάθηκε αρχικά το 1976 από τον Peter Chen [13] και στη συνέχεια έγινε ευρέως αποδεκτή από την επιστημονική κοινότητα.

Η Οντότητα (entity) είναι ένα αντικείμενο ενδιαφέροντος στον πραγματικό κόσμο το οποίο ξεχωρίζει από τα υπόλοιπα όπως π.χ. ένα συγκεκριμένο άτομο, εταιρεία ή γεγονός [13]. Ο Τύπος Οντότητας είναι μια συλλογή από χαρακτηριστικά ή ιδιότητες που περιγράφουν μία οντότητα όπως π.χ. φοιτητής, κατάσταση, υπάλληλος. Η Ιδιότητα της Οντότητας (attribute of an entity) περιγράφει ένα χαρακτηριστικό της οντότητας όπως π.χ. όνομα, ηλικία, διεύθυνση. Η Συσχέτιση (relationship) είναι η

σύνδεση μεταξύ των οντοτήτων όπως π.χ. διδάσκει, εργάζεται, αποτελείται. Η ιδιότητα της Συσχέτισης (attribute of a relationship) περιγράφει ένα χαρακτηριστικό της συσχέτισης [10]. Επιπλέον στο μοντέλο Οντοτήτων – Συσχετίσεων υπάρχει και η έννοια της πληθικότητας (cardinality) που δηλώνει αν οι σχέσεις μεταξύ των οντοτήτων είναι 1 προς 1 ή 1 προς πολλά.

Η μεταφορά της περιγραφής των απαιτήσεων σε στοιχεία του μοντέλου οντοτήτων - συσχετίσεων στη δημοσίευση του Chen [13], γίνεται μέσα από μία συλλογιστική η οποία προϋποθέτει την εμπλοκή του ανθρώπινου παράγοντα στην αποσαφήνιση του ρόλου κάθε λέξης.

Το 1983 ο Peter Chen πρότεινε μια σειρά από κανόνες μετατροπής (translation rules) [14] η εφαρμογή των οποίων σε (αγγλικό) κείμενο διατυπωμένο σε φυσική γλώσσα, μπορούσε να δημιουργήσει το διάγραμμα οντοτήτων – συσχετίσεων του κειμένου αυτού. Η ιδέα αυτή προήλθε από την αναγκαιότητα να μπορούν οι προδιαγραφές μιας βάσης δεδομένων να διατυπωθούν ως ανάλυση απαιτήσεων σε φυσική γλώσσα από ενδιαφερόμενους χωρίς τεχνικές γνώσεις και στη συνέχεια να υπάρχει η δυνατότητα εύκολης μετατροπής τους σε τύπους δεδομένων και δεδομένα της βάσης. Ο Peter Chen πρότεινε 11 κανόνες και τα επόμενα χρόνια προτάθηκαν σε δημοσιεύσεις περισσότεροι, καθώς και τρόποι για υπέρβαση κάποιων περιορισμών [18].

Οι κανόνες βασίζονται κυρίως στον καθορισμό κάθε στοιχείου του μοντέλου οντοτήτων – συσχετίσεων μέσα από τη γραμματική της κάθε πρότασης. Εντοπίζονται δηλαδή τα μέρη του λόγου τα οποία μεταφράζονται σε στοιχεία του διαγράμματος του μοντέλου. Σε πρώτη προσέγγιση τα ουσιαστικά μεταφράζονται σε τύπους οντοτήτων, τα ονόματα σε οντότητες, τα επίθετα σε ιδιότητες οντοτήτων, τα μεταβατικά ρήματα σε σχέσεις και τα επιρρήματα σε ιδιότητες σχέσεων. Σε αυτούς τους κανόνες προστίθενται άλλοι που υπερισχύουν σε συγκεκριμένες περιπτώσεις και άλλοι που προκύπτουν από τις προτάσεις τύπου AN X Σχέση1 Y ΤΟΤΕ X Σχέση2 Y, όπου X και Y είναι μέρη του λόγου [14].

Πριν από τη μετατροπή των μερών του λόγου σε διάγραμμα του οντοτήτων – συσχετίσεων η πρόταση αναλύεται λογικά, χωρίζεται σε τμήματα όπου είναι δυνατόν και ακολούθως εφαρμόζονται οι κανόνες του μοντέλου σε κάθε τμήμα. Η διαδικασία αυτή δεν είναι αυτοματοποιημένη και απαιτεί εποπτεία, ειδικά όταν πρόκειται για σύνθετες προτάσεις.

Κατά τη διάρκεια των επόμενων χρόνων μέχρι και σήμερα έχουν δημοσιευθεί εργασίες που αφορούν στην αυτόματη μετατροπή κειμένου διατυπωμένου σε φυσική γλώσσα σε διάγραμμα οντοτήτων – συσχετίσεων και κατ' επέκταση σε σχήματα και δεδομένα βάσεων δεδομένων [21] [38] [45] [18] [1]. Στις δημοσιεύσεις επιχειρείται μια προσέγγιση σε αυτοματοποιημένο σύστημα μετατροπής, ωστόσο σε καμία δεν περιγράφεται ένα πλήρως αυτοματοποιημένο σύστημα. Κατά την αναζήτηση που έγινε κατά τη διάρκεια της διατριβής δεν βρέθηκαν δημοσιευμένες αντίστοιχες εργασίες για κείμενα στην ελληνική γλώσσα.

2.5 Επεξεργασία φυσικής γλώσσας για την Ελληνική γλώσσα

Στην αναζήτηση που διεξήχθη στα πλαίσια της διατριβής για έρευνα, εργασίες, προϊόντα και υπηρεσίες σχετικά με την επεξεργασία της φυσικής γλώσσας για την Ελληνική γλώσσα, τα αποτελέσματα ήταν περιορισμένα σε σχέση με άλλες γλώσσες. Οι εταιρείες Google και Amazon προτείνουν να χρησιμοποιηθεί αυτόματος μεταφραστής για να μετατρέψει το ελληνικό κείμενο σε αγγλικό και στη

συνέχεια να χρησιμοποιηθούν στο αγγλικό κείμενο τα εργαλεία επεξεργασίας φυσικής γλώσσας που διαθέτουν.

Στον τομέα της συντακτικής ανάλυσης της φυσικής γλώσσας στην Ελλάδα έχει γίνει αρκετή έρευνα και έχουν αναπτυχθεί προϊόντα και υπηρεσίες από φορείς όπως το Ινστιτούτο Επεξεργασίας λόγου (ΙΕΛ) [58] , την Ομάδα Επεξεργασίας Φυσικής Γλώσσας του Οικονομικού Πανεπιστημίου Αθηνών [7] και το Εργαστήριο Επεξεργασίας Φυσικής Γλώσσας του Εθνικού Μετσόβιου Πολυτεχνείου [36] . Η δουλειά που έχει γίνει αφορά κυρίως στην αναγνώριση των ελληνικών λέξεων στα κείμενα, την άρση της ασάφειας των λέξεων στις προτάσεις και τη συντακτική ανάλυση των προτάσεων. Επίσης έχει γίνει δουλειά στη στατιστική επεξεργασία ελληνικών κειμένων και στη δημιουργία γλωσσικών πόρων στην Ελληνική γλώσσα. Σε επίπεδο σημασιολογίας έχουν αναπτυχθεί υπηρεσίες εντοπισμού και κατηγοριοποίησης οντοτήτων όπως η αναγνώριση τεχνικών όρων σε συγκεκριμένους τομείς και η αναγνώριση ονομάτων προσώπων και οργανισμών. Τέλος έχουν γίνει προσπάθειες σύνθεσης κειμένου με βάση πληροφορίες που προέρχονται από ερωτήματα σε βάσεις δεδομένων όπως σχεσιακές βάσεις για οντολογίες OWL [6] [28] .

Στην αναζήτηση που έγινε κατά τη διάρκεια της διατριβής, δεν βρήκαμε εφαρμογές για την αναπαράσταση των σχέσεων μεταξύ των οντοτήτων και ολοκληρωμένα συστήματα ερωταποκρίσεων τα οποία να προσφέρονται ως υπηρεσίες ή προϊόντα στην Ελληνική γλώσσα. Βρήκαμε μία χρήση του μοντέλου οντοτήτων – συσχετίσεων στη σχεδίαση και οργάνωση μιας λεξικολογικής βάσης δεδομένων για την Ελληνική γλώσσα [40] . Τα τελευταία χρόνια υπάρχουν διάφορες προσεγγίσεις στο θέμα σε εργασίες σε ακαδημαϊκό επίπεδο. Μία από τις τελευταίες είναι το διδακτορικό “NATURAL LANGUAGE INTERACTION WITH SEMANTIC WEB ONTOLOGIES” [27] στα πλαίσια του οποίου γίνεται αναπαράσταση εγγράφων με μορφή οντολογιών στη γλώσσα OWL. Οι πληροφορίες λαμβάνονται από τα έγγραφα με μεθόδους μηχανικής μάθησης και εξόρυξης δεδομένων με αυτόματα και ημιαυτόματα εξαγωγή. Επίσης μελετήθηκε και βελτιώθηκε ένα σύστημα παραγωγής κειμένων σε φυσική γλώσσα υψηλού επιπέδου. Οι γλώσσες που υποστηρίζει είναι Αγγλικά και Ελληνικά.

2.6 Συντακτικό της γλώσσας με κανόνες BNF και DCG

2.6.1 Περιγραφή γραμματικής με κανόνες BNF (Backus-Naur Form)

Ο συνηθισμένος τρόπος για να γίνει ακριβής ο ορισμός μιας γλώσσας, είτε πρόκειται για μια φυσική γλώσσα είτε για μια γλώσσα προγραμματισμού, είναι μέσω μιας συλλογής κανόνων που ονομάζεται «γραμματική». Οι κανόνες μιας γραμματικής καθορίζουν ποιες συμβολοσειρές από λέξεις ή σύμβολα είναι έγκυρες προτάσεις της γλώσσας. Επιπλέον, η γραμματική γενικά αναλύει την πρόταση σε μια δομή που καθιστά τη σημασία της πιο σαφή. Μια θεμελιώδης κλάση της γραμματικής είναι η γραμματική χωρίς περιεχόμενο (context-free grammar, CFG). Η BNF (Backus-Naur Form) είναι μια τεχνική συμβολισμού (μετασύνταξη) για γραμματικές χωρίς συμπραζόμενα (CFG), που χρησιμοποιείται για να περιγράψει τη σύνταξη μιας γλώσσας της πληροφορικής, όπως οι γλώσσες προγραμματισμού υπολογιστών. Εφαρμόζεται όπου χρειάζονται ακριβείς περιγραφές γλωσσών, για παράδειγμα σε επίσημους ορισμούς γλωσσών, σε εγχειρίδια, ή σε βιβλία για θεωρία γλωσσών προγραμματισμού.

Κάθε κανόνας της BNF ορίζεται σύμφωνα με την ακόλουθη μορφή:

<σύμβολο> ::= έκφραση

όπου το <σύμβολο> είναι ένα μη τερματικό (non-terminal) ενώ η έκφραση αποτελείται από μια ή περισσότερες ακολουθίες συμβόλων. Τα σύμβολα που δεν εμφανίζονται ποτέ στα αριστερά ονομάζονται τερματικά (terminals). Κάθε σύμβολο στη BNF βρίσκεται μέσα στους χαρακτήρες <> είτε εμφανίζεται στην αριστερή είτε στη δεξιά πλευρά του κανόνα. Τα σύμβολα που εμφανίζονται στην αριστερή πλευρά είναι μη τερματικά. Η έκφραση μπορεί να περιέχει τερματικά και μη τερματικά σύμβολα. Οι ακολουθίες συμβόλων στην έκφραση χωρίζονται μεταξύ τους με μια κάθετο, '|', που δείχνει επιλογή. Ένα τερματικό σύμβολο μπορεί να είναι ένα λεκτικό (literal) όπως "+" ή "function" ή μία κλάση από λεκτικά (literals) όπως integer.

Για παράδειγμα η τυπική γραμματική μιας αριθμητικής έκφρασης που περιλαμβάνει τις πράξεις της πρόσθεσης και του πολλαπλασιασμού φυσικών αριθμών σε BNF μορφή είναι η εξής:

```
<expr> ::= <term> "+" <expr>
        | <term>
<term> ::= <factor> "*" <term>
        | <factor>
<factor> ::= "(" <expr> ")"
          | <natural_number>
<natural_number> ::= "0"|"1"|"2"|"3"|...
```

Παραδείγματα εκφράσεων που αναγνωρίζει η γραμματική είναι τα εξής: 1) 5+(14*8), 2) (5*8)+12

Με τους κανόνες BNF μπορεί να οριστεί το συντακτικό τόσο μιας γλώσσας προγραμματισμού, όσο και μιας φυσικής γλώσσας.

Υπάρχουν πολλές επεκτάσεις και παραλλαγές της αρχικής BNF, κάποιες από αυτές είναι αυστηρά ορισμένες, όπως η Εκτεταμένη Μορφή BNF (Extended Backus–Naur Form, EBNF) και η Επαυξημένη Μορφή BNF (Augmented Backus–Naur Form, ABNF). Στα επόμενα θα χρησιμοποιείται η BNF με την προσθήκη του συμβόλου [] το οποίο σημαίνει προαιρετικό.

2.6.2 Περιγραφή γραμματικής με κανόνες DCG (definite clause grammar)

Οι Γραμματικές Οριστικών Προτάσεων (definite clause grammar, DCG) είναι μια γενίκευση των γραμματικών χωρίς συμφραζόμενα (CFG) και οι κανόνες που αφορούν στις DCG αναφέρονται ως «Κανόνες Γραμματικής». Με τη γραμματική οριστικών προτάσεων (DCG) μπορεί να εκφραστεί η γραμματική μιας φυσικής ή τυπικής γλώσσας και χρησιμοποιείται κυρίως σε προγράμματα Prolog [54]. Ονομάζεται γραμματική οριστικών προτάσεων γιατί αναπαριστά τη γραμματική ως μια συλλογή από οριστικές προτάσεις (definite clause) σε λογική πρώτης τάξης (first order logic).

Οι Γραμματικές Οριστικών Προτάσεων επεκτείνουν τις Context-free γραμματικές με τους ακόλουθους τρόπους [19]:

- Ένα μη τερματικό σύμβολο μπορεί να είναι οποιοσδήποτε όρος Prolog που μπορεί να κληθεί.
- Ένα τερματικό σύμβολο μπορεί να είναι οποιοσδήποτε όρος Prolog. Για τη διάκριση των τερματικών από τα μη τερματικά σύμβολα, μια σειρά από ένα ή περισσότερα τερματικά

σύμβολα γράφεται μέσα σε έναν κανόνα γραμματικής ως λίστα Prolog. Μια κενή ακολουθία γράφεται ως κενή λίστα '[]'. Εάν τα τερματικά σύμβολα είναι χαρακτήρες, αυτές οι λίστες μπορούν να γραφτούν ως συμβολοσειρές. Μια κενή ακολουθία γράφεται ως κενή λίστα ('[]' ή '""').

- Επιπλέον συνθήκες, με τη μορφή Prolog κώδικα, μπορούν να συμπεριληφθούν στη δεξιά πλευρά ενός κανόνα γραμματικής και είναι γραμμένες μέσα σε αγκύλες ('{' και '}').
- Η αριστερή πλευρά ενός κανόνα γραμματικής αποτελείται από ένα μη τερματικό σύμβολο, ακολουθούμενο προαιρετικά από μια ακολουθία τερματικών (επίσης γραμμένο ως λίστα Prolog).
- Εναλλακτικά σύμβολα μπορούν να τοποθετηθούν στη δεξιά πλευρά ενός κανόνα γραμματικής, χρησιμοποιώντας το ';' (ερωτηματικό) ως χαρακτήρα διαχωρισμού.
- Το θαυμαστικό '!' μπορεί να συμπεριληφθεί στη δεξιά πλευρά ενός κανόνα γραμματικής, όπως σε μια πρόταση Prolog. Το θαυμαστικό δεν χρειάζεται να περικλείεται σε αγκύλες. Το ίδιο ισχύει για δομές ελέγχου. Όλα τα άλλα ενσωματωμένα κατηγορήματα που δεν περικλείονται σε αγκύλες χειρίζονται ως μη τερματικά σύμβολα.
- Τα πρόσθετα ορίσματα των μη τερματικών παρέχουν τις έννοιες των δομών (όπως parse trees) σε κανόνες γραμματικής. Καθώς τα μη τερματικά σύμβολα «εξάγονται» με την αντιστοίχιση με τους με τους κανόνες γραμματικής, οι δομές αναπτύσσονται προοδευτικά κατά τη διάρκεια της διαδικασίας ενοποίησης (unification).
- Τα πρόσθετα ορίσματα των μη τερματικών μπορούν επίσης να παράσχουν μια γενική διαχείριση της εξάρτησης από τα συμφραζόμενα (context dependency) για να μεταφέρουν ελέγχους και πληροφορία συμφραζομένων (contextual).

Παράδειγμα

Ο παρακάτω BNF κανόνας στο οποίο το `<noun>` είναι ένα μη τερματικό σύμβολο και το «μαθητής» τερματικό σύμβολο,

```
<noun> ::= "μαθητής"
```

γράφεται σε σύνταξη DCG ως εξής:

```
noun --> [μαθητής].
```

Παρόμοια ο παρακάτω BNF κανόνας παραγωγής που αποτελείται μόνο από μη τερματικά σύμβολα

```
<sentence> ::= <noun_phrase> <verb_phrase> (ή <Πρόταση> ::= <Ονομαστική_Φράση>  
<Ρηματική_Φράση>)
```

γράφεται σε DCG ως

```
sentence --> noun_phrase, verb_phrase.
```

Με τον τρόπο αυτόν μπορούμε να δηλώσουμε τα τερματικά και τα μη τερματικά σύμβολα της γραμματικής. Παράδειγμα:

```
sentence --> noun_phrase, verb_phrase.
```

```
noun_phrase-->article,noun.
```



```
verb_phrase-->verb,noun_phrase.
```

```
noun-->[μαθητής];[βιβλίο].
```

```
article-->[ο].
```

```
verb-->[διαβάζει].
```

Με αυτό τον τρόπο η αναπαράσταση των γραμματικών κανόνων είναι περιγραφική και ταυτόχρονα μπορεί να ενσωματωθεί στον κώδικα της Prolog.

Οι φράσεις μιας γραμματικής οριστικών προτάσεων μπορούν να δεχτούν επιπλέον ορίσματα γεγονόσ που κάνει τις DCGs ικανές να εκφράσουν μια επέκταση των γραμματικών χωρίς συμφραζόμενα, τις κατηγορικές γραμματικές (attributed grammars) ή γραμματικές ιδιοτήτων. Οι κατηγορικές γραμματικές επιτρέπουν την προσκόλληση ιδιοτήτων στα σύμβολα της γραμματικής και την ύπαρξη σημασιολογικών κανόνων, οι οποίοι ορίζουν τις τιμές των ιδιοτήτων και πιθανά περιορισμούς σε αυτές. Οι γραμματικές ιδιοτήτων χρησιμοποιούνται ευρύτατα στην κατασκευή μεταγλωττιστών και επιτρέπουν, μέσω των DCG, να παράγεται από το συντακτικό δένδρο επιπλέον πληροφορία [57].

```
sentence( s( NP, VP ) ) -->
    noun_phrase( NP ), verb_phrase( VP ).
noun_phrase( np( A, N ) ) -->
    article( A ), noun( N ).
verb_phrase( vp( V, NP ) ) -->
    verb( V ), noun_phrase( NP ).
noun( n(μαθητής) ) --> [μαθητής].
noun( n(βιβλίο) ) --> [βιβλίο].
article( art(ο) ) --> [ο].
verb( v(διαβάζει) ) --> [διαβάζει].
```

2.7 Web-Programming με την SWI-Prolog

Μία από τις γλώσσες στις οποίες αναπτύσσονται εφαρμογές τεχνητής νοημοσύνης και υπολογιστικής γλωσσολογίας είναι η Prolog. Η Prolog βασίζεται στη λογική πρώτης τάξης (first-order logic) και είναι γλώσσα δηλωτικού προγραμματισμού που σημαίνει ότι η λογική του προγράμματος εκφράζεται με όρους και σχέσεις τα οποία αναπαρίστανται ως κανόνες και γεγονότα.

Μία από τις εκδόσεις της Prolog είναι η SWI-Prolog η οποία εκτός από τα χαρακτηριστικά της γλώσσας προγραμματισμού Prolog, παρέχει επιπλέον προγραμματιστικές δυνατότητες για δημιουργία web interface από την ίδια την εφαρμογή. Μέσα από τον κώδικα της εφαρμογής με την ενσωμάτωση βιβλιοθηκών μπορεί να δημιουργηθεί ένας web server και ένας HTTP Request Handler. Αυτό σημαίνει ότι η ίδια εφαρμογή με το ξεκίνημά της εκκινεί ένα web server ο οποίος κάνει "listen" σε συγκεκριμένη θύρα. Επιπλέον ο HTTP Request Handler μπορεί να δέχεται αιτήματα τύπου GET και POST και να

δημιουργεί και να επιστέφει ιστοσελίδες ή δεδομένα. Οι δυνατότητες αυτές παρέχουν ευχρηστία και ταχύτητα στις web εφαρμογές και ελαχιστοποιούν την ανάγκη για συντήρησή τους.

2.8 Σχετική δουλειά

Η δουλειά σχετικά με τα συστήματα ερωταποκρίσεων ως μέσο πρόσβασης σε βάσεις δεδομένων έχει ξεκινήσει από τη δεκαετία του '70 σε συστήματα κλειστού πεδίου (closed domain). Η πιο πρόσφατη δουλειά περιλαμβάνει συστήματα που χρησιμοποιούν επεξεργασία φυσικής γλώσσας (NLP) για πρόσβαση σε βάσεις δεδομένων αξιοποιώντας το μοντέλο Οντοτήτων Συσχετίσεων [14] ή επιχειρούν να μετατρέψουν ένα ερώτημα φυσικής γλώσσας σε μια αντίστοιχη λογική μορφή αξιοποιώντας σημασιολογικές συσχετίσεις μεταξύ της λεξικολογικής αναπαράστασης και των ιδιοτήτων βάσης δεδομένων στον λανθάνοντα χώρο (latent space) [6]. Στην κατεύθυνση αυτή ο Badia [8] έχει εισάγει μια γλώσσα επερωτήσεων με γενικευμένους ποσοτικοποιητές με σκοπό να γεφυρώσει το χάσμα μεταξύ των ερωτημάτων σε φυσική γλώσσα και των ερωτήσεων σε βάσεις δεδομένων. Όμως, παρά το γεγονός ότι οι ερωτήσεις μετασχηματίζονται σε λογική μορφή που ισοδυναμεί με ερώτημα (query) σε σχεσιακή βάση δεδομένων, δεν προσφέρουν απαντήσεις σε φυσική γλώσσα.

Επιπλέον, υπάρχει μια ισχυρή κοινότητα πίσω από τα συστήματα απάντησης ερωτήσεων που χρησιμοποιούν την αγγλική γλώσσα. Για παράδειγμα, το CLEF question answering track [42] αξιολογεί παρόμοια συστήματα τα οποία επιτρέπουν την πρόσβαση σε πληροφορίες χρησιμοποιώντας ερωτήσεις σε φυσική γλώσσα. Δεν υπάρχει όμως αντίστοιχο σύστημα αξιολόγησης για την Ελληνική γλώσσα.

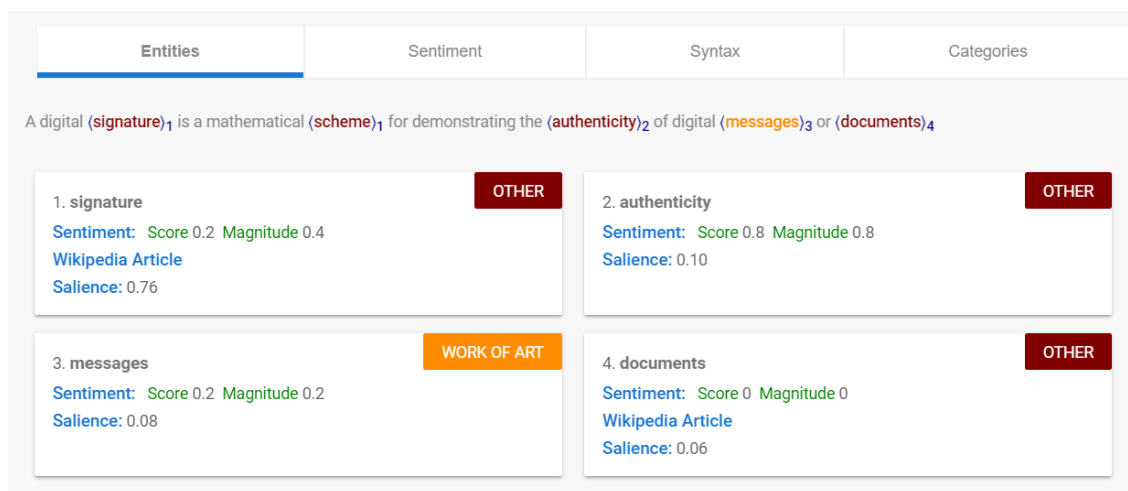
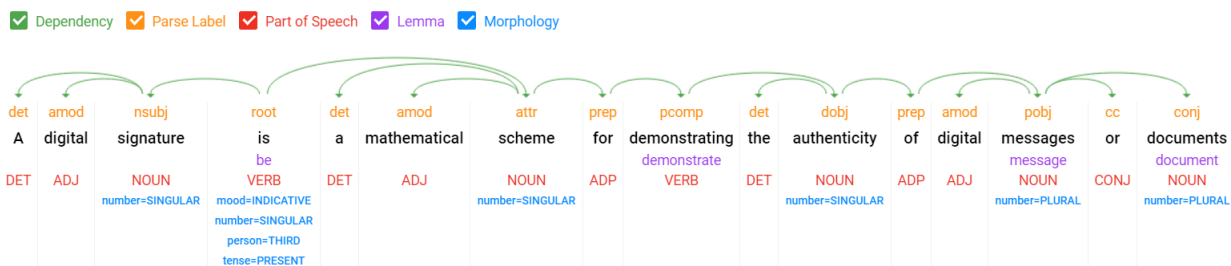
Σχετικά με την επεξεργασία φυσικής γλώσσας στα Ελληνικά, έχει γίνει περιορισμένη δουλειά σε σχέση με την αντίστοιχη στην Αγγλική. Για παράδειγμα έχουν αναπτυχθεί στελεχωτές (stemmers) για τον εντοπισμό του θέματος των λέξεων με την αφαίρεση των καταλήξεων. Το Ινστιτούτο Επεξεργασίας Λόγου έχει δημιουργήσει ένα σύνολο εργαλείων [39], [44] που βασίζονται τόσο σε αλγόριθμους μηχανικής μάθησης, όσο και σε κανόνες για επεξεργασία φυσικής γλώσσας για τα Ελληνικά. Ωστόσο, τα εργαλεία αυτά δεν υποστηρίζουν την απάντηση ερωτήματος και χρησιμοποιούνται κυρίως για εξαγωγή περίληψης κειμένου. Παρόλο που τα εργαλεία αυτά συμβάλλουν στην επεξεργασία της Ελληνικής γλώσσας, καλύπτουν μόνο συγκεκριμένες υποενότητες του συστήματος μας. Στην εργασία [31] έχει αναπτυχθεί ένα σύστημα ερωταποκρίσεων με τη χρήση του μοντέλου Οντοτήτων – Συσχετίσεων.

Στη συνέχεια παρατίθενται κάποια από τα online εργαλεία επεξεργασίας φυσικής γλώσσας που ήταν ελεύθερα ή εμπορικά διαθέσιμα κατά την εκπόνηση της διατριβής.

2.8.1 Ανάλυση κειμένου από Cloud Natural Language της Google

Η Google προσφέρει εμπορικά μέσω του Google Cloud μία εφαρμογή φυσικής γλώσσας η οποία αναλύει τη δομή και τις έννοιες ενός κειμένου. Διατίθεται μέσω ενός REST API, χρησιμοποιεί μοντέλα μηχανικής μάθησης και είναι διαθέσιμο (Ιανουάριος 2018) για την αγγλική και άλλες 9 γλώσσες αλλά όχι για τα Ελληνικά. Σύμφωνα με τη Google μπορεί να εξάγει πληροφορίες για ανθρώπους, τοποθεσίες, εκδηλώσεις κλπ που αναφέρονται μέσα σε έγγραφα κειμένου, άρθρα και blogs. Μπορεί να χρησιμοποιηθεί για κατανόηση συναισθήματος σχετικά με ένα προϊόν στα μέσα κοινωνικής δικτύωσης

ή για την κατανόηση των προθέσεων των πελατών [16] . Δοκιμάσαμε την εφαρμογή για την αγγλική μετάφραση της πρότασης «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα για την απόδειξη της αυθεντικότητας ψηφιακών μηνυμάτων ή εγγράφων» δηλαδή την πρόταση “A digital signature is a mathematical scheme for demonstrating the authenticity of digital messages or documents”. Το σύστημα ανέλυσε την πρόταση συντακτικά σχηματίζοντας το δέντρο συντακτικής ανάλυσης σε οριζόντια γραμμή δίνοντας παράλληλα πληροφορίες για την κλιτική τάξη των λέξεων της πρότασης όπως φαίνεται στο Σχήμα 5. Οι ετικέτες που εμφανίζονται αναφέρονται στην αντίστοιχη σελίδα του Google Cloud Natural Language API [29] .



Σχήμα 5: Η ανάλυση με την εφαρμογή Cloud Natural Language της Google [16] για την πρόταση: “A digital signature is a mathematical scheme for demonstrating the authenticity of digital messages or documents”

Αντίστοιχες εφαρμογές παρέχει η Microsoft με το Microsoft Azure [30] με υποστήριξη της Ελληνικής Γλώσσας και η Amazon με το Amazon Comprehend [4] χωρίς υποστήριξη της Ελληνικής γλώσσας.

2.8.2 Ανάλυση κειμένου από το Stanford NLP

Το πανεπιστήμιο του Stanford διεξάγει πολυετή και σημαντική έρευνα στον τομέα της επεξεργασία της φυσικής γλώσσας. Έχει δημιουργήσει πολλά εργαλεία ανάλυσης της φυσικής γλώσσας όπως statistical NLP, deep learning NLP, και rule-based NLP τα οποία διαθέτει δωρεάν. Οι γλώσσες που υποστηρίζονται είναι Αγγλικά, Ισπανικά Κινέζικα και άλλες γλώσσες αλλά όχι Ελληνικά.

Δώσαμε στον Stanford Parser [48] για ανάλυση την πρόταση “*A digital signature is a mathematical scheme for demonstrating the authenticity of digital messages or documents*”. Ο Parser δημιούργησε το δέντρο συντακτικής ανάλυσης της πρότασης παρέχοντας πληροφορίες για την κλιτική τάξη των λέξεων και τη δομή της πρότασης όπως φαίνεται στο *Σχήμα 6*.

Tagging

A/DT digital/JJ signature/NN is/VBZ a/DT mathematical/JJ
scheme/NN for/IN demonstrating/VBG the/DT authenticity/NN of/IN
digital/JJ messages/NNS or/CC documents/NNS

Parse

```
(ROOT
 (S
  (NP (DT A) (JJ digital) (NN signature))
  (VP (VBZ is)
   (NP
    (NP (DT a) (JJ mathematical) (NN scheme))
    (PP (IN for)
     (S
      (VP (VBG demonstrating)
       (NP
        (NP (DT the) (NN authenticity))
        (PP (IN of)
         (NP (JJ digital) (NNS messages)
          (CC or)
          (NNS documents))))))))))
```

Universal dependencies

```
det(signature-3, A-1)
amod(signature-3, digital-2)
nsubj(scheme-7, signature-3)
cop(scheme-7, is-4)
det(scheme-7, a-5)
amod(scheme-7, mathematical-6)
root(ROOT-0, scheme-7)
mark(demonstrating-9, for-8)
acl(scheme-7, demonstrating-9)
det(authenticity-11, the-10)
dobj(demonstrating-9, authenticity-11)
case(messages-14, of-12)
amod(messages-14, digital-13)
nmod(authenticity-11, messages-14)
cc(messages-14, or-15)
conj(messages-14, documents-16)
```

Universal dependencies, enhanced

```
det(signature-3, A-1)
amod(signature-3, digital-2)
nsubj(scheme-7, signature-3)
cop(scheme-7, is-4)
det(scheme-7, a-5)
amod(scheme-7, mathematical-6)
root(ROOT-0, scheme-7)
mark(demonstrating-9, for-8)
acl(scheme-7, demonstrating-9)
det(authenticity-11, the-10)
dobj(demonstrating-9, authenticity-11)
case(messages-14, of-12)
amod(messages-14, digital-13)
nmod:of(authenticity-11, messages-14)
cc(messages-14, or-15)
nmod:of(authenticity-11, documents-16)
conj:or(messages-14, documents-16)
```

Σχήμα 6: Η ανάλυση με τον Stanford Parser [48] της πρότασης:

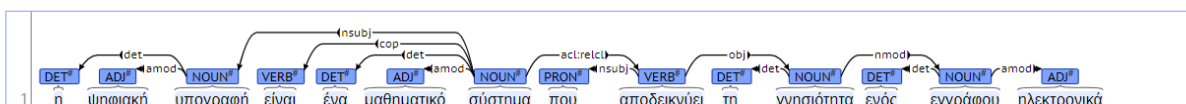
“A digital signature is a mathematical scheme for demonstrating the authenticity of digital messages or documents”

2.8.3 Ανάλυση κειμένου από το Ινστιτούτο Επεξεργασίας του Λόγου (ΙΕΛ)

Το Ινστιτούτο Επεξεργασίας του Λόγου (ΙΕΛ) είναι ελληνικό ερευνητικό ινστιτούτο που έχει σκοπό την ανάπτυξη και προώθηση της Γλωσσικής Τεχνολογίας στην Ελλάδα. Διεξάγει βασική και εφαρμοσμένη έρευνα στον τομέα της Γλωσσικής Τεχνολογίας. Έχει αναπτύξει μια σειρά από εργαλεία γλωσσικής τεχνολογίας για την ελληνική γλώσσα. Δοκιμάσαμε κάποια από αυτά τα οποία είναι διαθέσιμα ως online υπηρεσίες για να αναλύσουμε την πρόταση «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά».

Η υπηρεσία ΙΕΛ dependency parser χρησιμοποιεί τα εργαλεία επεξεργασίας της φυσικής γλώσσας (ILSP NLP) [23] για να αναλύσει συντακτικά κείμενα γραμμένα στην Ελληνική γλώσσα και να δημιουργήσει αναπαραστάσεις συμβατές με το “Universal Dependencies schema”. Η υπηρεσία δημιουργεί επισημειώσεις μερών του λόγου και λημμάτων στις λέξεις της πρότασης. Επίσης δημιουργεί τα δέντρα εξαρτήσεων (dependency trees) με τη χρήση του “dependency parser”. Διαθέτει επιλογή για έξοδο του δέντρου της συντακτικής ανάλυσης σε διάφορες μορφές. Δώσαμε για ανάλυση στο συντακτικό αναλυτή την πρόταση «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά». Δημιούργησε και επέστρεψε το δέντρο συντακτικής ανάλυσης της πρότασης σε οριζόντια γραμμή, καθώς και τη λίστα με τις λέξεις της πρότασης παρέχοντας πληροφορίες για το μέρος του λόγου την κλιτική τάξη κάθε λέξης. Το δέντρο συντακτικής ανάλυσης που δημιουργήθηκε φαίνεται στο Σχήμα 7. Οι ετικέτες που χρησιμοποιούνται προέρχονται από το Universal Dependencies [49] [17] .

ILSP NLP Dependency Parser Results



Σχήμα 7: Η ανάλυση με από την υπηρεσία ΙΕΛ dependency parser [23] της πρότασης:
«Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά»

3. Περιγραφή του συστήματος ερωταποκρίσεων με τη χρήση εξαγωγής πληροφοριών από κείμενα

3.1 Κύρια χαρακτηριστικά του συστήματος μας

Τα κύρια χαρακτηριστικά του συστήματος ερωταποκρίσεων που αναπτύξαμε στα πλαίσια της διατριβής, συνοψίζονται παρακάτω:

1. Επεξεργασία φυσικής γλώσσας (Ελληνικά) και ανάλυση κειμένου (ερωτήσεις και προτάσεις).
2. Αναπαράσταση της συντακτικής δομής του κειμένου σε λογική (κατηγορήματα).
3. Εξαγωγή σημασιολογικής πληροφορίας και δημιουργία διαγράμματος οντοτήτων – συσχετίσεων για την αναπαράσταση της πληροφορίας με μορφή σημασιολογικού δέντρου.
4. Αποθήκευση των δεδομένων της αναπαράστασης του διαγράμματος οντοτήτων – συσχετίσεων ως σύνολο γεγονότων στη βάση γνώσης.
5. Ανάλυση και σημασιολογική αναπαράσταση της δομής του κειμένου των ερωτήσεων ως σημασιολογικό δέντρο.
6. Σύγκριση των σημασιολογικών δέντρων των ερωτήσεων με τα δέντρα που είναι αποθηκευμένα στη βάση με σκοπό την εξαγωγή απαντήσεων.
7. Δημιουργία απαντήσεων σε φυσική γλώσσα από τα δεδομένα των σημασιολογικών δέντρων.

Για τη λειτουργία του συστήματος έχουν αναπτυχθεί μεταξύ των άλλων τα παρακάτω γλωσσικά εργαλεία σε κώδικα Prolog:

1. Λεκτικός αναλυτής ο οποίος παράγει τις λέξεις σε όλες τις κλιτικές τάξεις χρησιμοποιώντας τα δεδομένα του λεξικού.
2. Συντακτικός αναλυτής ο οποίος ελέγχει και αναγνωρίζει τη συντακτική δομή των προτάσεων.
3. Επισημειωτής μερών του λόγου (POS Tagger) ο οποίος προσδιορίζει το μέρος του λόγου και την κλιτική τάξη κάθε λέξης.
4. Συλλαβιστής ο οποίος χωρίζει τις λέξεις σε συλλαβές σύμφωνα με τους κανόνες συλλαβισμού.
5. Τονιστής, ο οποίος τονίζει τις λέξεις σύμφωνα με τους κανόνες τονισμού.

Επιπλέον στην εφαρμογή έχει αναπτυχθεί κώδικας Prolog ο οποίος:

1. Υλοποιεί τη μετατροπή κειμένου σε στοιχεία του μοντέλου οντοτήτων – συσχετίσεων εφαρμόζοντας τους κανόνες του μοντέλου.
2. Μετατρέπει τα κείμενα των ερωτήσεων και των προτάσεων σε σημασιολογικά δέντρα.
3. Συγκρίνει τα σημασιολογικά δέντρα ερωτήσεων και αποθηκευμένων πληροφοριών και εντοπίζει τις πληροφορίες των απαντήσεων.
4. Μετατρέπει πληροφορίες σε κείμενο φυσικής γλώσσας και συντάσσει καλοσηματισμένες απαντήσεις στην Ελληνική γλώσσα λαμβάνοντας υπόψιν και στοιχεία από το κείμενο της ερώτησης.

3.2 Περιορισμοί και επεκτάσεις

Το σύστημα διαχειρίζεται τα άρθρα, ουσιαστικά, επίθετα, ρήματα και επιρρήματα. Επίσης τις ερωτηματικές αντωνυμίες «ποιος» και «τι» και τον σύνδεσμο «και». Δε διαχειρίζεται τα υπόλοιπα μέρη του λόγου. Τα ρήματα που αναγνωρίζει βρίσκονται στον ενεστώτα της ενεργητικής φωνής σε έγκλιση οριστική. Μπορεί επίσης να αναγνωρίσει γραμματικά ρήματα σε παθητική φωνή (σε οριστική ενεστώτα).

Η σημασιολογική αναπαράσταση των κειμένων γίνεται για τα μέρη του λόγου για τα οποία υπάρχουν διατυπωμένοι κανόνες του Μοντέλου Οντοτήτων - Συσχετίσεων (ουσιαστικά, επίθετα, ρήματα και επιρρήματα). Τα υπόλοιπα μέρη του λόγου (π.χ άρθρα) αγνοούνται.

Οι ερωτήσεις που επεξεργάζεται είναι της μορφής «Ποιος» και «Τι». Δεν διαχειρίζεται ερωτήσεις της μορφής «Πόσοι», «Ποιον», «Ποιου», «Πώς», «Πότε», «Γιατί».

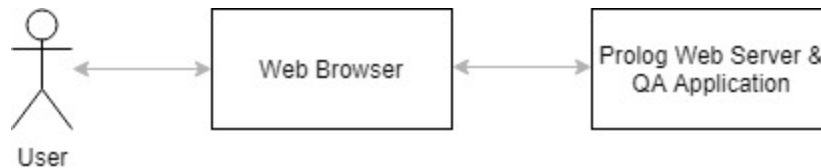
Παρά τους περιορισμούς, το σύστημα μπορεί να αναγνωρίσει τη σημασιολογική δομή ενός μεγάλου εύρους κειμένων, όπως φαίνεται στα παραδείγματα των επόμενων κεφαλαίων.

Το σύστημα έχει αναπτυχθεί με γνώμονα την επεκτασιμότητα χωρίς σημαντικές, ή με μηδενικές αλλαγές στα δομικά του στοιχεία. Η γραμματική μπορεί να επεκταθεί με την προσθήκη νέων γεγονότων και κατηγορημάτων τα οποία θα προσθέτουν νέες λέξεις και κανόνες γραμματικής της γλώσσας. Ο χειρισμός του χρόνου της φωνής και των εγκλίσεων στα ρήματα γίνεται με την προσθήκη επιπλέον ορισμάτων στα σχετικά κατηγορήματα και την προσθήκη μικρών τμημάτων κώδικα τα οποία θα λαμβάνουν υπόψιν τα συγκεκριμένα ορίσματα και θα επεκτείνουν τις υπάρχουσες δυνατότητες παραγωγής της κλιτικής τάξης των ρημάτων. Στη σημασιολογική αναπαράσταση μπορούν να προστεθούν νέοι κανόνες για τη διαχείριση επιπλέον μερών του λόγου όπως π.χ. μετοχές. Για τη δημιουργία νέων τύπων απαντήσεων χρειάζεται η προσθήκη κώδικα που θα τις διαχειρίζεται. Π.χ. για να απαντηθεί το ερώτημα «Πόσοι τρέχουν;» θα πρέπει πρώτα να απαντηθεί το ερώτημα «Ποιοι τρέχουν;» και στη συνέχεια να γίνει απαρίθμηση των οντοτήτων.

Μια λίστα με τις πιθανές επεκτάσεις του συστήματος παρατίθεται στο Κεφάλαιο 7.

4. Αρχιτεκτονική του συστήματος

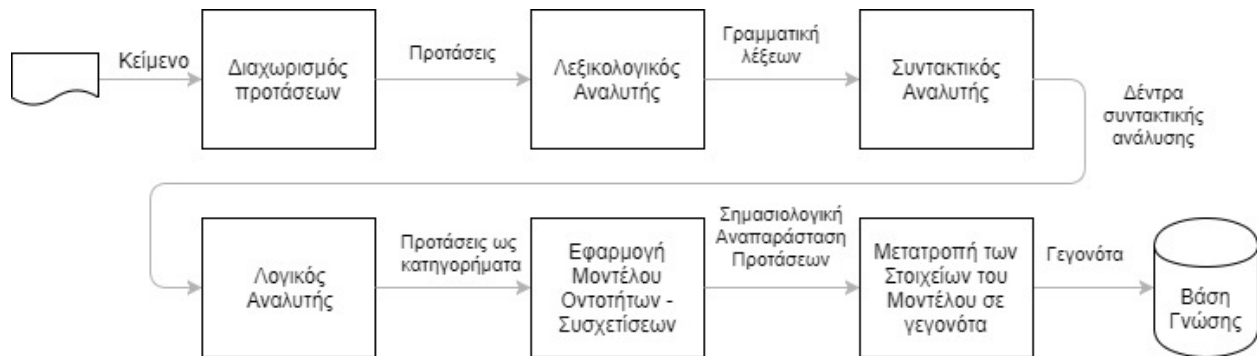
Η εφαρμογή που αναπτύχθηκε στα πλαίσια της διατριβής λειτουργεί με τη μορφή client - server. Ο χρήστης μέσα από έναν web browser συνδέεται στον web server της εφαρμογής. Ο web server όπως και ο HTTP Request Handler είναι τμήματα της εφαρμογής μας η οποία έχει τη δυνατότητα να κάνει “listen” και να δέχεται αιτήματα από τον browser σε συγκεκριμένο port, να τα διαχειρίζεται και να επιστρέφει απαντήσεις σε μορφή σελίδων html (Σχήμα 8).



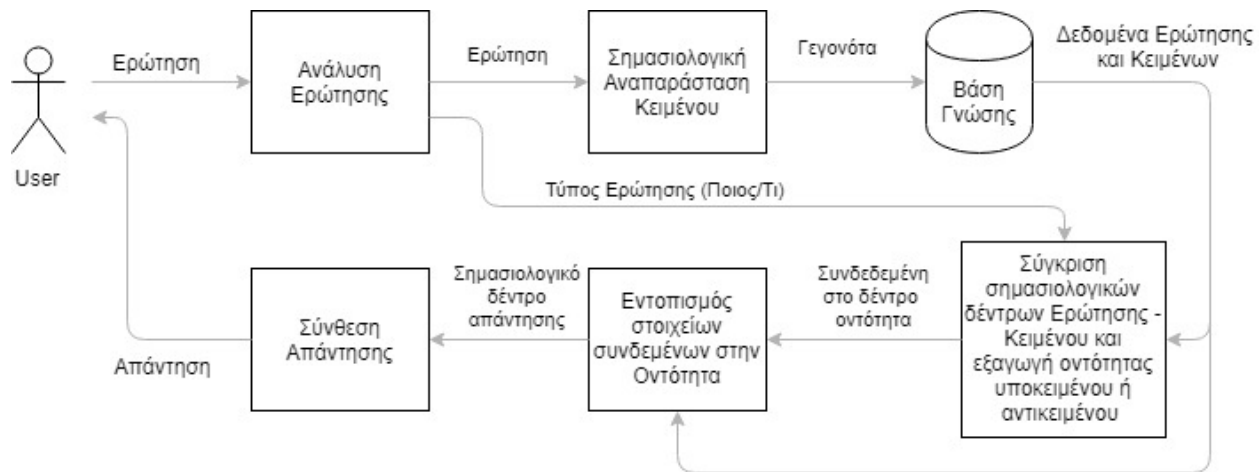
Σχήμα 8. Η τοπολογία Client - Server του Συστήματος Ερωταποκρίσεων

Ο χρήστης μπορεί να εισάγει στην εφαρμογή κείμενα για επεξεργασία και αποθήκευση της δομής τους και ερωτήματα προς απάντηση.

Η ανάλυση των κειμένων που εισάγονται γίνεται ακολουθώντας τα στάδια που φαίνονται στο Σχήμα 9. Τα κείμενα που εισάγονται στην εφαρμογή αρχικά διαχωρίζονται σε προτάσεις. Στη συνέχεια αναγνωρίζονται οι λέξεις και γενικότερα οι λεκτικές μονάδες των προτάσεων. Αφού καθοριστεί το μέρος του λόγου και η μορφή της κάθε λέξης, ελέγχονται σύμφωνα με τη γραμματική της γλώσσας από το Συντακτικό Αναλυτή ο οποίος παράγει το δέντρο συντακτικής ανάλυσης (parse tree) για κάθε πρόταση. Το δέντρο αυτό στη συνέχεια αναπαριστάνεται με λογική μορφή ως κατηγορημα της Prolog από τον Λογικό Αναλυτή. Ακολούθως, στη λογική αναπαράσταση της συντακτικής δομής της πρότασης εφαρμόζονται οι κανόνες του Μοντέλου Οντοτήτων – Συσχετίσεων και παράγονται τα στοιχεία του μοντέλου δηλαδή οι οντότητες, οι ιδιότητες και οι σχέσεις που περιέχει η πρόταση. Τα στοιχεία αυτά συνθέτουν το σημασιολογικό δέντρο αναπαράστασης της πρότασης το οποίο μετατρέπεται σε μια σειρά γεγονότων ενός κατηγορήματος το οποίο χρησιμοποιείται για τη αναπαράσταση της σημασιολογίας της πρότασης.



Σχήμα 9. Τα στάδια ανάλυσης των κειμένων από την εφαρμογή



Σχήμα 10. Τα στάδια για τη δημιουργία της απάντησης στο ερώτημα του χρήστη

Η εφαρμογή δέχεται ερωτήσεις στις οποίες αναζητάει απάντηση αξιοποιώντας τα δεδομένα που βρίσκονται αποθηκευμένα στη βάση της. Τα στάδια για τη δημιουργία των απαντήσεων στα ερωτήματα που εισάγει ο χρήστης φαίνονται στο Σχήμα 10. Αρχικά η ερώτηση αναλύεται ως προς τον τύπο της, ώστε με βάση αυτόν να καθοριστεί ο τύπος της ζητούμενης απάντησης. Μετά γίνεται επεξεργασία της ερώτησης με τον ίδιο τρόπο όπως γίνεται και στα κείμενα, στη συνέχεια η σημασιολογική της αναπαράσταση αποθηκεύεται στη βάση γνώσης. Ακολούθως συγκρίνεται το σημασιολογικό δέντρο της ερώτησης με το σημασιολογικό δέντρο όλων των πληροφοριών που είναι αποθηκευμένες στη βάση. Εφόσον το σημασιολογικό δέντρο της ερώτησης ταυτιστεί με ένα τμήμα του σημασιολογικού δέντρου των καταχωρημένων πληροφοριών της βάσης, αναζητούνται οι οντότητες οι οποίες είναι συνδεδεμένες στο τμήμα αυτό. Το ποια συνδεδεμένη οντότητα θα χρησιμοποιηθεί ως απάντηση εξαρτάται από το είδος της ερώτησης και ανάλογα με αυτό, μπορεί να είναι το υποκείμενο ή το αντικείμενο μιας σχέσης. Στη συνέχεια εντοπίζονται οι ιδιότητες της οντότητας αυτής όπως και άλλες οντότητες που ενδέχεται να συνδέονται μαζί της, οι οποίες τη χαρακτηρίζουν ή τη συγκεκριμενοποιούν και μαζί με αυτήν δημιουργούν το σημασιολογικό δέντρο της απάντησης. Αυτό το σημασιολογικό δέντρο χρησιμοποιείται για τη σύνθεση της απάντησης σε φυσική γλώσσα που θα δοθεί στο χρήστη. Για τη σύνθεσή της χρησιμοποιούνται τα λεξικά, οι κανόνες των κλίσεων των λέξεων και το συντακτικό της γλώσσας όπως επίσης ο τύπος και η διατύπωση της ερώτησης. Η απάντηση επιστρέφεται ως απόκριση του συστήματος στο χρήστη και εμφανίζεται στον web browser.

Στο επόμενο κεφάλαιο περιγράφονται αναλυτικά όλα τα δομικά στοιχεία που αποτελούν την εφαρμογή και ο τρόπος που υλοποιείται τόσο η σημασιολογική αναπαράσταση των κειμένων, όσο και η παροχή απαντήσεων στα ερωτήματα.

5. Αναλυτική περιγραφή του συστήματος

Για την υλοποίηση της εφαρμογής αναπτύχθηκαν και συνενώθηκαν ξεχωριστές δομικές μονάδες οι οποίες εκτελούν τις ακόλουθες λειτουργίες:

- Ανάλυση της πρότασης και δημιουργία δέντρου συντακτικής ανάλυσης, λογική αναπαράσταση του δέντρου συντακτικής ανάλυσης, εξαγωγή οντοτήτων και συσχετίσεων σύμφωνα με το Μοντέλο Οντοτήτων – Συσχετίσεων, μετατροπή των στοιχείων του μοντέλου σε γεγονότα κατηγορημάτων και αποθήκευση της αναπαράστασης της σημασιολογικής δομής της πρότασης στη βάση γνώσης.
- Αναζήτηση πληροφοριών στη βάση γνώσης σχετικά με την ερώτηση του χρήστη με σύγκριση σημασιολογικών δέντρων, σύνθεση απάντησης από το δέντρο σημασιολογικής δομής της απάντησης και
- User interface της εφαρμογής που τρέχει ως client σε web browser.

Ο τρόπος σχεδίασης και οι τεχνικές υλοποίησης κάθε τμήματος περιγράφεται τις ενότητες που ακολουθούν.

5.1 Λεξικολογική, γραμματική και συντακτική ανάλυση

Για να αναγνωριστούν οι προτάσεις που εισάγονται στην εφαρμογή και για να παραχθούν οι απαντήσεις στα Ελληνικά, θα πρέπει να οριστούν με σαφήνεια το λεξικό και οι γραμματικοί κανόνες της γλώσσας. Η επισημείωση των μερών του λόγου (Part Of Speech tagging, POS tagging) βασίζεται σε προκαθορισμένα λεξικά και γραμματικούς κανόνες (rule-based POS Tagging). Στις περιπτώσεις που χρειάζεται αποσαφήνιση του ρόλου μιας λέξης μέσα στην πρόταση, χρησιμοποιούνται οι γραμματικοί και συντακτικοί κανόνες οι οποίοι τη συνδέουν με τις γειτονικές της. Για παράδειγμα ένα επίθετο που χαρακτηρίζει ένα ουσιαστικό πρέπει να βρίσκεται στο ίδιο γένος, αριθμό και πτώση με αυτό.

Οι λέξεις της Ελληνικής γλώσσας που αναγνωρίζονται από την εφαρμογή, είναι καταχωρημένες ως γεγονότα σε ξεχωριστό λεξικό για κάθε μέρος του λόγου. Κάθε λεξικό περιέχει τις πληροφορίες που χαρακτηρίζουν γραμματικά τις λέξεις που περιέχει, όπως τον τρόπο κλίσης, τον τονισμό και το γένος για τα ουσιαστικά ή το πρόσωπο για τα ρήματα. Τα λεξικά παρέχουν έναν ευέλικτο τρόπο για τη διεύρυνση των δυνατοτήτων αναγνώρισης κειμένων της εφαρμογής, προσθέτοντας νέες λέξεις, χωρίς αλλαγές στον κώδικα της εφαρμογής. Οι κλίσεις των λέξεων δημιουργούνται με βάση τα γεγονότα στο αρχείο κάθε λεξικού συνδυαζόμενες με τις αντίστοιχες καταλήξεις από τα γεγονότα στο αρχείο των καταλήξεων για τα κλιτά μέρη του λόγου. Για τα ανώμαλα ρήματα, των οποίων το θέμα αλλάζει κατά την κλίση, υπάρχει ξεχωριστό λεξικό από αυτό των υπολοίπων ρημάτων.

5.1.1 Λεξικά

Για κάθε μέρος του λόγου που υποστηρίζει η εφαρμογή υπάρχει το αντίστοιχο λεξικό σε αρχείο Prolog. Τα λεξικά που χρησιμοποιούνται αναφέρονται στον Πίνακα 3. Επιπλέον, χρησιμοποιούνται λεξικά για τις καταλήξεις των ουσιαστικών, των επιθέτων και των ρημάτων τα οποία βρίσκονται στα αντίστοιχα αρχεία Prolog όπως φαίνεται στον Πίνακα 4.

Μέρος του λόγου	Αρχείο
Άρθρα	articles.pl
Ουσιαστικά	nouns.pl
Κύρια ονόματα	proper_nouns.pl
Επίθετα	adjectives.pl
Ρήματα	verbs.pl
Ανώμαλα ρήματα	verbs_irregular.pl
Άρθρα	articles.pl
Επιρήματα	adverbs.pl
Ερωτηματικές αντωνυμίες	interrogative_pronouns.pl
Σύνδεσμοι	conjunctions.pl

Πίνακας 3. Τα μέρη του λόγου και τα αντίστοιχα αρχεία Prolog με τα λεξικά που τα περιέχουν

Καταλήξεις	Αρχείο
Ουσιαστικών	noun-suffixes.pl
Επιθέτων	adjective-suffixes.pl
Ρημάτων	verb-suffixes.pl

Πίνακας 4. Οι καταλήξεις για τα μέρη του λόγου και τα αντίστοιχα αρχεία Prolog τις περιέχουν

5.1.1.1 Λεξικό ουσιαστικών

Το λεξικό των ουσιαστικών βρίσκεται στο αρχείο της **nouns.pl** Prolog ως το κατηγορημα **gr_noun_lexicon/4** το οποίο έχει τη μορφή:

gr_noun_lexicon(Ρίζα, Γένος, Κωδικός_Κατάληξης, Τονισμός).

όπου

- «Ρίζα» είναι το μέρος του ουσιαστικού (θέμα) που δε μεταβάλεται κατά την κλίση.
- «Γένος» είναι το γένος του ουσιαστικού με τιμές “arseniko”, “thiliko”, “oydetero”.
- «Κωδικός_Κατάληξης» είναι ένα Id με το οποίο γίνεται αναφορά σε προτάσεις γεγονότα του κατηγορηματος **gr_noun_suffix/5** που βρίσκονται στο αρχείο “**noun-suffixes.pl**”. Κάθε γεγονός περιέχει τις καταλήξεις για την αντίστοιχη κατηγορία ουσιαστικών σε κάθε αριθμό και πτώση.

- «Τονισμός» είναι ο αριθμός της συλλαβής στην οποία τονίζεται η λέξη, μετρώντας από την τελευταία με τιμές 1 για τη λήγουσα, 2 για την παραλήγουσα και 3 για την προπαραλήγουσα.

Παράδειγμα: `gr_noun_lexicon('μαθητ', arseniko, 10, 1)`.

Οι καταλήξεις των ουσιαστικών ανάλογα με το είδος, τον αριθμό και την πτώση δίνονται από το κατηγορήμα `gr_noun_suffix/5` το οποίο έχει τη μορφή:

`gr_noun_suffix`(Κωδικός_Κατάληξης, Γένος, Αριθμός, Πτώση, Κατάληξη).

Όπου

- «Κωδικός_Κατάληξης» είναι το Id με το οποίο γίνεται αναφορά σε κάθε κατάληξη.
- «Γένος» είναι το γένος του ουσιαστικού με τιμές “arseniko”, “thiliko”, “oydetero” της κατάληξης.
- «Αριθμός» είναι ο αριθμός (ενικός/πληθυντικός) με τιμές “enikos”, “plithintikos” της κατάληξης.
- «Πτώση» είναι η πτώση της κατάληξης (ονομαστική, γενική, αιτιατική) με τιμές “onomastiki”, “geniki”, “aitiatiki”. Κλιτική πτώση δεν υπάρχει γιατί δε χρησιμοποιείται στους τύπους των κειμένων της διατριβής, αλλά μπορεί να προστεθεί ως ένα ακόμα γεγονός.
- «Κατάληξη» είναι η κατάληξη στην οποία αναφέρεται το συγκεκριμένου Id, για συγκεκριμένα «Γένος», «Αριθμός» και «Πτώση».

Παράδειγμα: `gr_noun_suffix(10, arseniko, enikos, onomastiki, 'ης')`.

Το λεξικό των ουσιαστικών χρησιμοποιείται τόσο στον προσδιορισμό της κλιτικής τάξης των ουσιαστικών, όσο και στην παραγωγή των κλίσεων των ουσιαστικών. Για τον προσδιορισμό της μορφολογίας των ουσιαστικών έχει δημιουργηθεί το κατηγορήμα `gr_noun21/8` ενώ για τη σύνθεση το κατηγορήμα `mk_gr_noun/5`.

Στον κώδικα της εφαρμογής το κατηγορήμα

`gr_noun21/8`

`gr_noun21(X, F, Genos, Arithmos, Ptosi, Tropos, NounType, Word, L)`.

χρησιμοποιώντας τα γεγονότα του λεξικού και των καταλήξεων των ουσιαστικών επιστρέφει για κάθε μορφή του ουσιαστικού που αποτελεί στοιχείο της λίστας `Word`, το γένος, τον αριθμό και την πτώση του. Η αναζήτηση μπορεί να περιοριστεί καλώντας το κατηγορήμα με συγκεκριμένες τιμές στο γένος, τον αριθμό, την πτώση ή συνδυασμό τους.

Η έξοδος από το κατηγορήμα με την παρακάτω κλήση

`gr_noun21(X, F, Genos, Arithmos, Ptosi, Tropos, NounType, [τάξης], [])`.

θα επιστρέψει:

`F = ταξ(X)`, % αναπαράσταση ονομαστικής φράσης, π.χ.
 % 1) για «ήσυχη τάξη» το `F` θα έχει το «ταξ(ησυχ)»,
 % 2) για «λέξη κλειδί» το `F` θα έχει το «λεξ(κλειδ)».
`Genos = thiliko,`

```
Arithmos = enikos,  
Ptosi = geniki,  
Tropos = 7,  
NounType = noun .
```

Τα ορίσματα **X** και **F** χρησιμοποιούνται για τη λογική αναπαράσταση της δομής της πρότασης.

Για την παραγωγή των κλίσεων των ουσιαστικών χρησιμοποιείται το κατηγορήμα:

mk_gr_noun/5

mk_gr_noun(Riza, Genos, Arithmos, Ptosi, Word)

το οποίο παράγει την κλίση του ουσιαστικού για δεδομένη ρίζα (θέμα) και για συγκεκριμένα γένος, αριθμό και πτώση, χρησιμοποιώντας τα γεγονότα του λεξικού των ουσιαστικών και των καταλήξεων. Με αυτόν τον τρόπο παράγεται η μορφή κάθε ουσιαστικού κατά την κλίση του.

Η έξοδος από το κατηγορήμα με την παρακάτω κλήση

```
mk_gr_noun('εγκαταστας', thiliko, plithintikos, geniki, Word)
```

θα επιστρέψει:

```
Word = εγκαταστάσεων.
```

Το λεξικό των κύριων ονομάτων βρίσκεται στο αρχείο **proper_nouns.pl** της Prolog ως το κατηγορήμα **gr_proper_noun_lexicon/4** το οποίο έχει τη μορφή:

gr_proper_noun_lexicon(Ρίζα, Γένος, Κωδικός_Κατάληξης, Τονισμός).

Τα ορίσματα είναι ίδια με το λεξικό των ουσιαστικών. Οι καταλήξεις λαμβάνονται από το αρχείο **noun-suffixes.pl** στο οποίο βρίσκονται και οι καταλήξεις των ουσιαστικών. Η διαφορά είναι ότι στην κλήση των κυρίων ονομάτων δεν έχει οριστεί πληθυντικός αριθμός.

Παράδειγμα: `gr_proper_noun_lexicon('Γιωργ', arseniko, 4, 2) .`

5.1.1.2 Λεξικό επιθέτων

Το λεξικό των επιθέτων έχει αποθηκευτεί στη βάση γνώσης της Prolog ως το γεγονός **gr_adject_lexicon/3** το οποίο έχει τη μορφή:

gr_adject_lexicon(Ρίζα, Κωδικός_Κατάληξης, Τονισμός).

όπου

- «Ρίζα» είναι το μέρος του επιθέτου (θέμα) που δε μεταβάλλεται κατά την κλίση.
- «Κωδικός_Κατάληξης» είναι ένα Id με το οποίο γίνεται αναφορά σε προτάσεις γεγονότα του κατηγορήματος **gr_adject_suffix/5** που βρίσκονται στο αρχείο **adjective-suffixes.pl**. Κάθε γεγονός περιέχει για την αντίστοιχη κατηγορία επιθέτων τις καταλήξεις τους για κάθε γένος, αριθμό και πτώση.

- «Τονισμός» είναι ο αριθμός της συλλαβής στην οποία τονίζεται η λέξη μετρώντας από την τελευταία με τιμές 1 για τη λήγουσα, 2 για την παραλήγουσα και 3 για την προπαραλήγουσα.

Παράδειγμα: `gr_adject_lexicon('μεγαλ', 1, 2)`.

Το κατηγορήμα ***gr_adject_suffix/5*** με τις καταλήξεις των επιθέτων έχει την εξής μορφή:

gr_adject_suffix(Κωδικός_Κατάληξης, Γένος, Αριθμός, Πτώση, Κατάληξη).

Όπου

- «Κωδικός_Κατάληξης» είναι το Id με το οποίο γίνεται αναφορά σε κάθε κατάληξη.
- «Γένος» είναι το γένος του επιθέτου με τιμές “arseniko”, “thiliko”, “oydetero” το οποίο έχει τη συγκεκριμένη κατάληξη.
- «Αριθμός» είναι ο αριθμός της κατάληξης με τιμές “enikos”, “plithintikos”.
- «Πτώση» είναι η πτώση της κατάληξης (ονομαστική, γενική, αιτιατική) με τιμές “onomastiki”, “geniki”, “aitiatiki”. Κλιτική πτώση δεν υπάρχει γιατί δε χρησιμοποιείται στους τύπους των κειμένων της διατριβής, αλλά μπορεί να προστεθεί ως ένα ακόμα γεγονός.
- «Κατάληξη» είναι η κατάληξη του συγκεκριμένου Id, στα αναφερόμενα «Γένος», «Αριθμός» και «Πτώση».

Παράδειγμα: `gr_adject_suffix(1, arseniko, enikos, onomastiki, 'ος')`.

Το λεξικό των επιθέτων χρησιμοποιείται τόσο στον προσδιορισμό της κλιτικής τάξης των επιθέτων, όσο και στην παραγωγή των κλίσεών τους. Για τον προσδιορισμό της μορφολογίας των επιθέτων έχει δημιουργηθεί το κατηγορήμα ***gr_adjective/8*** ενώ για τη σύνθεσή τους το κατηγορήμα ***mk_gr_adj/5***.

Στον κώδικα της εφαρμογής το κατηγορήμα

gr_adjective/8

gr_adjective (X, F, Genos, Arithmos, Ptsi, Tropos, Word, L).

χρησιμοποιώντας τα γεγονότα του λεξικού και των καταλήξεων των επιθέτων, επιστρέφει για κάθε μορφή του επιθέτου που αποτελεί στοιχείο της λίστας Word, το γένος, τον αριθμό και την πτώση του. Η αναζήτηση μπορεί να περιοριστεί καλώντας το κατηγορήμα με συγκεκριμένες τιμές για το γένος, τον αριθμό, την πτώση ή συνδυασμό τους.

Η παρακάτω κλήση στο κατηγορήμα

```
gr_adjective (X, F, Genos, Arithmos, Ptsi, Tropos, ['πράσινης'], []).
```

θα επιστρέψει:

```
X = [πρασιν],
F = πρασιν([], [adjective, πρασιν, thiliko, enikos, geniki, 1]),
Genos = thiliko,
Arithmos = enikos,
```

```
Ptosi = geniki,  
Tropos = 1.
```

Τα ορίσματα **X** και **F** χρησιμοποιούνται για τη λογική αναπαράσταση της δομής της πρότασης.

Για την παραγωγή των κλίσεων των επιθέτων χρησιμοποιείται το κατηγορημα:

mk_gr_adj/5

mk_gr_adj (Riza, Genos, Arithmos, Ptosi, Word)

το οποίο παράγει την κλίση του επιθέτου για δεδομένη ρίζα (θέμα) στα συγκεκριμένα γένος, αριθμό και πτώση, χρησιμοποιώντας τα γεγονότα του λεξικού των επιθέτων και των καταλήξεων. Με αυτό τον τρόπο παράγεται η μορφή κάθε επιθέτου κατά την κλίση του σε όλα τα γένη.

Η παρακάτω κλήση στο κατηγορημα

```
mk_gr_adj('ιδιωτικ', thiliko, plithintikos, aitiatiki, Word).
```

θα επιστρέψει:

```
Word = ιδιωτικές .
```

5.1.1.3 Λεξικό ρημάτων

Η εφαρμογή χρησιμοποιεί δύο λεξικά ρημάτων, ένα για τα ομαλά και ένα για τα ανώμαλα ρήματα.

Το λεξικό των ομαλών ρημάτων βρίσκεται στο αρχείο της Prolog **verbs.pl** ως το κατηγορημα **gr_verb_lexicon/4** το οποίο έχει τη μορφή:

gr_verb_lexicon(Ρίζα, Κωδικός_Κατάληξης, Μεταβατικότητα, Τονισμός).

όπου

- «Ρίζα» είναι το μέρος του ρήματος (θέμα) που δε μεταβάλεται κατά την κλίση.
- «Κωδικός_Κατάληξης» είναι ένα Id με το οποίο γίνεται αναφορά σε προτάσεις γεγονότα του κατηγορηματος **gr_verb_suffix/4** που βρίσκονται στο αρχείο **verb-suffixes.pl**. Κάθε γεγονός περιέχει για την αντίστοιχη κατηγορία ρημάτων τις καταλήξεις τους για κάθε αριθμό και πρόσωπο.
- «Μεταβατικότητα» είναι η μεταβατικότητα του ρήματος με τιμές “trans” για τα μεταβατικά και “intrans” για τα αμετάβαρα ρήματα.
- «Τονισμός» είναι ο αριθμός της συλλαβής στην οποία τονίζεται η λέξη μετρώντας από την τελευταία με τιμές 1 για τη λήγουσα και 2 για την παραλήγουσα και 3 για την προπαραλήγουσα.

Παράδειγμα: `gr_verb_lexicon('διδασκ', 1, trans, 2) .`

Το κατηγορημα **gr_verb_suffix/4** το οποίο βρίσκεται στο αρχείο **verb-suffixes.pl** περιέχει τις καταλήξεις των ρημάτων (ομαλών και ανώμαλων) και έχει την εξής μορφή:

gr_verb_suffix(Κωδικός_Κατάληξης, Αριθμός, Πρόσωπο, Κατάληξη).

όπου

- «Κωδικός_Κατάληξης» είναι το Id με το οποίο γίνεται αναφορά σε κάθε κατάληξη.
- «Αριθμός» είναι ο αριθμός της κατάληξης με τιμές “enikos”, “plithintikos”.
- «Πρόσωπο» είναι το πρόσωπο της κατάληξης με τιμές “proto”, “deytero” και “trito”.
- «Κατάληξη» είναι η κατάληξη του συγκεκριμένου Id, στα αναφερόμενα «Αριθμός» και «Πρόσωπο».

Παράδειγμα: `gr_verb_suffix(1,plithintikos,proto,'ουμε')`.

Το κατηγορημα μπορεί να επεκταθεί με την προσθήκη νέων ορισμάτων ώστε να περιέχει πληροφορία για τη φωνή των ρημάτων, εγκλίσεις, συζυγίες κατηγορίες κλπ.

Το λεξικό των ρημάτων χρησιμοποιείται τόσο στον προσδιορισμό της κλιτικής τάξης των ρημάτων, όσο και στην παραγωγή των κλίσεων τους. Για τον προσδιορισμό της μορφολογίας των ρημάτων έχει δημιουργηθεί το κατηγορημα **gr_verb2/5** ενώ για τη σύνθεση το κατηγορημα **mk_gr_verb/5**.

Στον κώδικα της εφαρμογής το κατηγορημα

gr_verb2/5

gr_verb2 (Riza,Arithmos,Prosopo,Trans,Word).

χρησιμοποιώντας τα γεγονότα του λεξικού και των καταλήξεων των ρημάτων, επιστρέφει για κάθε μορφή του ρήματος που αποτελεί στοιχείο της λίστας Word, τη ρίζα, τον αριθμό, το πρόσωπο και τη μεταβατικότητα του. Η αναζήτηση μπορεί να περιοριστεί καλώντας το κατηγορημα με συγκεκριμένες τιμές για τον αριθμό, το πρόσωπο, τη μεταβατικότητά ή συνδυασμό τους.

Η παρακάτω κλήση στο κατηγορημα

```
gr_verb2(Riza,Arithmos,Prosopo,Trans,Tropos,[περιγράφουν],[ ]).
```

θα επιστρέψει:

```
Riza = περιγραφ,  
Arithmos = plithintikos,  
Prosopo = trito,  
Trans = trans,  
Tropos = 1.
```

Για την παραγωγή των κλίσεων των ρημάτων χρησιμοποιείται το κατηγορημα:

mk_gr_verb/5

mk_gr_verb (Riza,Arithmos,Prosopo,Trans,Word)

το οποίο παράγει την κλίση του ρήματος για δεδομένη ρίζα (θέμα) και για συγκεκριμένο αριθμό και πρόσωπο, χρησιμοποιώντας τα γεγονότα του λεξικού των ρημάτων και των καταλήξεων. Με αυτό τον τρόπο παράγεται η μορφή κάθε ρήματος κατά την κλίση του σε όλα τα πρόσωπα.

Η παρακάτω κλήση στο κατηγορημα

```
mk_gr_verb('παρακολουθ', plithintikos, proto, Trans, Word) .
```

Θα επιστρέψει:

```
Trans = trans,  
Word = παρακολουθούμε .
```

Στα ανώμαλα ρήματα κατά την κλίση, αλλάζει εκτός από την κατάληξη και η ρίζα (θέμα) τους. Το λεξικό των ανώμαλων ρημάτων περιέχει τη ρίζα της λέξης σε όλες τις κλίσεις και έχει αποθηκευτεί στη βάση γνώσης της Prolog στο αρχείο **verbs_irregular.pl** ως το κατηγορημα **gr_iverb_lexicon/6** το οποίο έχει τη μορφή:

gr_iverb_lexicon (Κωδικός_Κατάληξης, Ρίζα, Ρήμα, Αριθμός, Πρόσωπο, Μεταβατικότητα, Τονισμός).

όπου

- «Κωδικός_Κατάληξης» είναι ένα Id με το οποίο γίνεται αναφορά σε προτάσεις γεγονότα του κατηγορήματος **gr_verb_suffix/4** που βρίσκονται στο αρχείο **verb-suffixes.pl**. Κάθε γεγονός περιέχει για την αντίστοιχη κατηγορία ρημάτων τις καταλήξεις τους για κάθε αριθμό και πρόσωπο.
- «Ρίζα» είναι το θέμα της λέξης σε κάθε κλίση.
- «Ρήμα» είναι η ρίζα στο τρίτο πρόσωπο ενικού με την οποία γίνεται αναφορά στο ρήμα ανεξάρτητα από την κλίση.
- «Μεταβατικότητα» είναι η μεταβατικότητα του ρήματος με τιμές “trans” για τα μεταβατικά και “intrans” για τα αμετάβαρα ρήματα.
- «Τονισμός» είναι ο αριθμός της συλλαβής στην οποία τονίζεται η λέξη μετρώντας από την τελευταία με τιμές 1 για τη λήγουσα και 2 για την παραλήγουσα και 3 για την προπαραλήγουσα.

Παράδειγμα:

```
gr_iverb_lexicon(10, 'ειμ', 'ειν', enikos, proto, trans, 2) .  
gr_iverb_lexicon(10, 'εισ', 'ειν', enikos, deytero, trans, 2) .  
gr_iverb_lexicon(10, 'ειν', 'ειν', enikos, trito, trans, 2) .
```

Η αναζήτηση για τα ρήματα γίνεται πρώτα στο λεξικό των ομαλών και αν δεν εντοπιστούν συνεχίζεται στο λεξικό των ανωμάτων ρημάτων.

5.1.1.4 Λεξικό άρθρων

Το λεξικό των άρθρων βρίσκεται στο αρχείο **articles.pl** της Prolog ως το κατηγορημα **gr_article_lexicon/4** το οποίο έχει τη μορφή:

gr_article_lexicon(Γένος, Αριθμός, Πτώση, Άρθρο).

όπου

- «Γένος» είναι το γένος του άρθρου με τιμές “arseniko”, “thiliko”, “oydetero”.
- «Αριθμός» είναι ο αριθμός του άρθρου με τιμές “enikos”, “plithintikos”.
- «Πτώση» είναι η πτώση του άρθρου με τιμές “onomastiki”, “geniki”, “aitiatiiki”.
- «Άρθρο» είναι το άρθρο στο συγκεκριμένο «Γένος», «Αριθμός» και «Πτώση».

Οι μορφές της κλίσης του άρθρου «ο» σε γένος αρσενικό είναι καταχωρημένες ως γεγονότα στο αρχείο **articles.pl** όπως φαίνεται παρακάτω:

```
gr_article_lexicon(arseniko, enikos, onomastiki, 'ο').  
gr_article_lexicon(arseniko, enikos, geniki, 'του').  
gr_article_lexicon(arseniko, enikos, aitiatiiki, 'τον').  
gr_article_lexicon(arseniko, enikos, aitiatiiki, 'το').  
gr_article_lexicon(arseniko, plithintikos, onomastiki, 'οι').  
gr_article_lexicon(arseniko, plithintikos, geniki, 'των').  
gr_article_lexicon(arseniko, plithintikos, aitiatiiki, 'τους').
```

Σημειώνεται ότι στην αιτιατική του ενικού το άρθρο μπορεί να έχει τις εναλλακτικές μορφές «το» ή «τον».

5.1.1.5 Λεξικό επιρρημάτων

Το λεξικό των επιρρημάτων βρίσκεται στο αρχείο **adverbs.pl** της Prolog ως το κατηγορήμα **gr_adverb_lexicon/2** το οποίο έχει τη μορφή:

gr_adverb_lexicon(Επίρρημα, Σχέση).

όπου

- «Επίρρημα» το συγκεκριμένο επίρρημα (π.χ. τώρα, πολύ, εδώ)
- «Σχέση» η σχέση που δηλώνει το συγκεκριμένο επίρρημα (χρονικό, ποσοτικό, τοπικό) με τιμές “xroniko”, “posotiko”, “topiko”.

Παράδειγμα:

```
gr_adverb_lexicon('τώρα', xroniko).  
gr_adverb_lexicon('πολύ', posotiko).  
gr_adverb_lexicon('εδώ', topiko).
```

5.1.1.6 Λεξικό συνδέσμων

Το λεξικό των συνδέσμων βρίσκεται στο αρχείο **conjunctions.pl** της Prolog ως το κατηγορήμα **gr_conjunction_lexicon/2** το οποίο έχει τη μορφή:

gr_conjunction_lexicon(Σύνδεσμος, Είδος).

όπου

- «Σύνδεσμος» ο συγκεκριμένος σύνδεσμος (π.χ. και, που)

- «Είδος» το είδος του συγκεκριμένου συνδέσμου με τιμές “symplektikos” και “eidikos”.

Παράδειγμα:

```
gr_conjunction_lexicon('και', symplektikos).
gr_conjunction_lexicon('που', eidikos).
```

5.1.1.7 Λεξικό ερωτηματικών αντωνυμιών

Το λεξικό των ερωτηματικών αντωνυμιών βρίσκεται στο αρχείο **interrogative_pronouns.pl** της Prolog ως το κατηγορήμα **gr_ipronoun_lexicon/4** το οποίο έχει τη μορφή:

gr_ipronoun_lexicon(Ερ_Αντωνυμία, Γένος, Αριθμός, Πτώση).

όπου

- «Ερωτ_Αντωνυμία» είναι η ερωτηματική αντωνυμία στο συγκεκριμένο «Γένος», «Αριθμός» και «Πτώση», π.χ. (ποιος, τι)
- «Γένος» είναι το γένος της ερωτηματικής αντωνυμίας με τιμές “arseniko”, “thiliko”, “oydetero”.
- «Αριθμός» είναι ο αριθμός της ερωτηματικής αντωνυμίας με τιμές “enikos”, “plithintikos”.
- «Πτώση» είναι η πτώση της ερωτηματικής αντωνυμίας.

Παράδειγμα:

```
gr_ipronoun_lexicon('ποιος', arseniko, enikos, onomastiki).
gr_ipronoun_lexicon('ποιου', arseniko, enikos, geniki).
gr_ipronoun_lexicon('ποιον', arseniko, enikos, aitiatiki).
gr_ipronoun_lexicon('ποιους', arseniko, plithintikos, onomastiki).
gr_ipronoun_lexicon('ποιων', arseniko, plithintikos, geniki).
gr_ipronoun_lexicon('ποιους', arseniko, plithintikos, aitiatiki).
```

5.1.2 Παραγωγή κλίσεων λέξεων

Με τη χρήση των δεδομένων από τα λεξικά και τις καταλήξεις μπορούν να δημιουργηθούν οι διαφορετικές μορφές των λέξεων ανάλογα με την κλίση τους. Για να δημιουργηθούν συνδυάζεται το άκλιτο μέρος της λέξης, δηλαδή το θέμα, το οποίο είναι καταχωρημένο στο λεξικό, με τις καταλήξεις που αντιστοιχούν σε κάθε κλίση, εφόσον το μέρος του λόγου στο οποίο ανήκει η λέξη κλίνεται. Με αυτόν τον τρόπο δημιουργούνται οι κλίσεις των ουσιαστικών, των επιθέτων και των ρημάτων. Για τα συγκεκριμένα μέρη του λόγου, εφαρμόζονται στη συνέχεια οι κανόνες τονισμού και δημιουργείται η λέξη στην τελική της τονισμένη μορφή. Τα άρθρα είναι καταχωρημένα στον αντίστοιχο πίνακα σε όλες τις κλίσεις τονισμένα. Τα άκλιτα μέρη του λόγου δηλαδή επιρρήματα, σύνδεσμοι και ερωτηματικές αντωνυμίες είναι καταχωρημένα στους αντίστοιχους πίνακες επίσης τονισμένα.

5.1.2.1 Κλίση ουσιαστικών

Με τη χρήση των λεξικών και των καταλήξεων αναλύονται λεξικολογικά τα κείμενα και παράγονται οι κλίσεις των λέξεων. Το ουσιαστικό «μαθητής» είναι αρσενικό με κατάληξη -ης. Στην ίδια κατηγορία ανήκουν τα ουσιαστικά «επιβάτης», «ναύτης» και άλλα. Το ουσιαστικό «μαθητής» έχει θέμα το «μαθητ», είναι αρσενικό με κωδικό κατάληξης 10 και τονίζεται στη λήγουσα. Οι πληροφορίες αυτές είναι καταχωρημένες με το ακόλουθο γεγονός στο αρχείο **nouns.pl**

```
gr_noun_lexicon('μαθητ', arseniko, 10, 1).
```

Στον πίνακα **noun-suffixes.pl** για τον κωδικό κατάληξης 10 είναι καταχωρημένες οι καταλήξεις του ουσιαστικού στις διάφορες πτώσεις και αριθμούς με τις παρακάτω γραμμές.

```
gr_noun_suffix(arseniko, enikos, onomastiki, 10, 'ης').  
gr_noun_suffix(arseniko, enikos, geniki, 10, 'η').  
gr_noun_suffix(arseniko, enikos, aitiatiki, 10, 'η').  
gr_noun_suffix(arseniko, plithintikos, onomastiki, 10, 'ες').  
gr_noun_suffix(arseniko, plithintikos, geniki, 10, 'ων').  
gr_noun_suffix(arseniko, plithintikos, aitiatiki, 10, 'ες').
```

Η παρακάτω κλίση στο κατηγορήμα

```
gr_noun21(X, F, Genos, Arithmos, Ptosi, Tropos, NounType, [μαθητές], []).
```

θα επιστρέψει:

```
F = μαθητ(X),  
Genos = arseniko,  
Arithmos = plithintikos,  
Ptosi = onomastiki,  
Tropos = 10,  
NounType = noun.
```

Όταν σε μια πρόταση εντοπιστεί η λέξη «μαθητές» και κληθεί το κατηγορήμα **gr_noun21/9** η εφαρμογή θα αναγνωρίσει ότι πρόκειται για αρσενικό ουσιαστικό στην ονομαστική πληθυντικού με ρίζα «μαθητ». Μετά την ταυτοποίησή της η λέξη «μαθητές» συνοδεύεται από τις πληροφορίες

Θέμα: μαθητ

Κατάληξη: -ες

Γένος: Αρσενικό

Αριθμός: Πληθυντικός

Πτώση: Ονομαστική

Κωδικός κατάληξης: 10

Μέρος του Λόγου: Ουσιαστικό

Οι πληροφορίες αυτές περιλαμβάνονται κωδικοποιημένες στη λίστα
[noun, μαθητ, arseniko, plithintikos, geniki, 10]
η οποία συνοδεύει τη λέξη ως *γραμματική της λέξης*.

Για την παραγωγή της κλίσης ενός ουσιαστικού σε συγκεκριμένο αριθμό και πτώση χρησιμοποιείται το κατηγορήμα

mk_gr_noun/5

mk_gr_noun(Riza, Genos, Arithmos, Ptosí, Word).

το οποίο συνδυάζοντας το θέμα με τις αντίστοιχες καταλήξεις, παράγει τις κλίσεις του ουσιαστικού για δεδομένο γένος, αριθμό και πτώση. Καλώντας το με ρίζα «μαθητ», και τους συνδυασμούς των τιμών για Genos=arseniko, Arithmos=enikos, plithintikos και Ptosí=onomastiki, geniki, aitiatiki, δημιουργείται ο Πίνακας 5 με τις κλίσεις του ουσιαστικού μαθητής.

Πτώση/Αριθμός	Ενικός	Πληθυντικός
Ονομαστική	μαθητ-ης	μαθητ-ες
Γενική	μαθητ-η	μαθητ-ων
Αιτιατική	μαθητ-η	μαθητ-ες

Πίνακας 5. Η κλίση του ουσιαστικού μαθητής όπως παράγεται από το κατηγορήμα *mk_gr_noun/5*

Οι λέξεις που δημιουργούνται από το συνδυασμό του θέματος και της κατάληξης δεν είναι τονισμένες. Οι κανόνες τονισμού εφαρμόζονται στο επόμενο στάδιο. Κάθε μορφή της λέξης που παρήχθηκε, συνοδεύεται από τις πληροφορίες της γραμματικής βάσει της οποίας προέκυψε (γραμματική της λέξης).

5.1.2.2 Κλίση επιθέτων

Χρησιμοποιώντας δεδομένα από το λεξικό των επιθέτων **adjectives.pl** και των καταλήξεων επιθέτων **adjective-suffixes.pl** δημιουργούνται οι κλίσεις των επιθέτων στα τρία γένη: αρσενικό, θηλυκό και ουδέτερο.

Το επίθετο «ψηφιακός» έχει κατάληξη –ος. Στην ίδια κατηγορία ανήκουν τα επίθετα «μεγάλος», «κρύος» και άλλα. Για το επίθετο «ψηφιακός» είναι καταχωρημένη η πληροφορία ότι το θέμα είναι το «ψηφιακ», έχει κωδικό κατάληξης 1 και τονίζεται στη λήγουσα, με το ακόλουθο γεγονός στο αρχείο **adjectives.pl**

```
gr_adject_lexicon('ψηφιακ',1,1) .
```

Στο αρχείο **adjective-suffixes.pl** υπάρχουν τα παρακάτω γεγονότα που ορίζουν τις καταλήξεις των επιθέτων με κωδικό κατάληξης 1 (στα οποία ανήκει το επίθετο «ψηφιακός») για κάθε γένος, αριθμό και πτώση.

```
gr_adject_suffix(arseniko,enikos,onomastiki,1,'ος') .  
gr_adject_suffix(arseniko,enikos,geniki,1,'ου') .  
gr_adject_suffix(arseniko,enikos,aitiatiki,1,'ο') .  
gr_adject_suffix(arseniko,plithintikos,onomastiki,1,'οι') .  
gr_adject_suffix(arseniko,plithintikos,geniki,1,'ων') .  
gr_adject_suffix(arseniko,plithintikos,aitiatiki,1,'ους') .
```

```

gr_adject_suffix(thiliko,enikos,onomastiki,1,'η').
gr_adject_suffix(thiliko,enikos,geniki,1,'ης').
gr_adject_suffix(thiliko,enikos,aitiatiki,1,'η').
gr_adject_suffix(thiliko,plithintikos,onomastiki,1,'ες').
gr_adject_suffix(thiliko,plithintikos,geniki,1,'ων').
gr_adject_suffix(thiliko,plithintikos,aitiatiki,1,'ες').

gr_adject_suffix(oydetero,enikos,onomastiki,1,'ο').
gr_adject_suffix(oydetero,enikos,geniki,1,'ου').
gr_adject_suffix(oydetero,enikos,aitiatiki,1,'ο').
gr_adject_suffix(oydetero,plithintikos,onomastiki,1,'α').
gr_adject_suffix(oydetero,plithintikos,geniki,1,'ων').
gr_adject_suffix(oydetero,plithintikos,aitiatiki,1,'α').

```

Η έξοδος από το κατηγορήμα με την παρακάτω κλήση

```
gr_adjective(X,F,Genos,Arithmos,Ptosi,Tropos,['ψηφιακών'],[]).
```

Θα επιστρέψει:

```

X = [ψηφιακ],
F = ψηφιακ([], [adjective, ψηφιακ, arseniko, plithintikos, geniki, 1]),
Genos = arseniko,
Arithmos = plithintikos,
Ptosi = geniki,
Tropos = 1.

```

Το κατηγορήμα θα σταματήσει την αναζήτηση λύσεων μόλις εντοπίσει την πρώτη, δηλαδή θα επιστρέψει το πρώτο γένος που ταυτίζεται με τη μορφή του επιθέτου και όχι τα υπόλοιπα. Αυτό είναι αρκετό στις περιπτώσεις που ελέγχεται αν η λέξη «ψηφιακών» συμπεριλαμβάνεται στο λεξικό. Το κατηγορήμα θα πρέπει όμως να καθορίζει αν η μορφή του επιθέτου που αναζητείται ταυτίζεται με τις κλίσεις του και στα υπόλοιπα γένη, αριθμούς και πτώσεις. Επειδή τα επίθετα χαρακτηρίζουν τα ουσιαστικά, το γένος του επιθέτου θα πρέπει να είναι το ίδιο με το γένος του ουσιαστικού.

Υπάρχουν περιπτώσεις που η αναζήτηση πρέπει να περιοριστεί σε συγκεκριμένο γένος, αριθμό ή πτώση. Αν για παράδειγμα ένα επίθετο προηγείται ενός ουσιαστικού και έχει προσδιοριστεί το γένος του ουσιαστικού π.χ. αρσενικό, τότε και το επίθετο πρέπει να είναι στο ίδιο γένος. Στην περίπτωση αυτή το κατηγορήμα **gr_adjective** θα κληθεί με Genos=arseniko και η αναζήτηση των κλίσεων του επιθέτου θα περιοριστεί στα αρσενικά. Αντίστοιχα περιορίζονται ο αριθμός και η πτώση του επιθέτου που αναζητείται.

Αν μέσα στην πρόταση υπάρχουν οι λέξεις «... ψηφιακών σημάτων ...» θα προσδιοριστεί ότι το γένος του ουσιαστικού είναι ουδέτερο, και θα κληθεί το κατηγορήμα με συγκεκριμένο γένος ως εξής:

```
gr_adjective(X,F,oydetero,Arithmos,Ptosi,Tropos,['ψηφιακών'],[]).
```

X = [ψηφιακ],
F = ψηφιακ([], [adjective, ψηφιακ, **oydetero**, plithintikos, geniki, 1]),
Genos = **oydetero**,
Arithmos = plithintikos,
Ptosi = geniki,
Tropos = 1.

Μετά την ταυτοποίησή της η λέξη «ψηφιακών» συνοδεύεται από τις πληροφορίες

Θέμα: ψηφιακ
Κατάληξη: -ων
Γένος: Ουδέτερο
Αριθμός: Πληθυντικός
Πτώση: Γενική
Κωδικός κατάληξης: 1
Μέρος του Λόγου: Επίθετο

Οι πληροφορίες αυτές περιλαμβάνονται κωδικοποιημένες στη λίστα
[adjective, ψηφιακ, oydetero, plithintikos, geniki, 1]
η οποία συνοδεύει τη λέξη ως *γραμματική της λέξης*.

Για την παραγωγή της κλίσης ενός επιθέτου σε συγκεκριμένο γένος, αριθμό και πτώση χρησιμοποιείται το κατηγορημα

mk_gr_adj/5

mk_gr_adj(Riza,Genos,Arithmos,Ptosi,Word).

το οποίο συνδυάζοντας το θέμα με τις αντίστοιχες καταλήξεις, παράγει τις κλίσεις του επιθέτου για δεδομένο γένος, αριθμό και πτώση. Καλώντας το με ρίζα «ψηφιακ», και τους συνδυασμούς των τιμών για Genos=Arseniko, Arithmos=enikos, plithintikos και Ptosi=onomastiki, geniki, aitiatiki, δημιουργείται ο Πίνακας 6 με τις κλίσεις του επιθέτου ψηφιακός.

Οι λέξεις που δημιουργούνται από το συνδυασμό του θέματος και της κατάληξης, όπως και στα ουσιαστικά, δεν είναι τονισμένες. Οι κανόνες τονισμού εφαρμόζονται στο επόμενο στάδιο. Κάθε μορφή της λέξης που παράχθηκε, συνοδεύεται από τις πληροφορίες της γραμματικής βάσει της οποίας προέκυψε.

Πτώση/Αριθμός	Ενικός	Πληθυντικός
<i>Αρσενικό</i>		
Ονομαστική	ψηφιακ-ος	ψηφιακ-οι
Γενική	ψηφιακ-ου	ψηφιακ-ων
Αιτιατική	ψηφιακ-ο	ψηφιακ-ους
<i>Θηλυκό</i>		
Ονομαστική	ψηφιακ-η	ψηφιακ-ες
Γενική	ψηφιακ-ης	ψηφιακ-ων
Αιτιατική	ψηφιακ-η	ψηφιακ-ες
<i>Ουδέτερο</i>		
Ονομαστική	ψηφιακ-ο	ψηφιακ-α
Γενική	ψηφιακ-ου	ψηφιακ-ων
Αιτιατική	ψηφιακ-ο	ψηφιακ-α

Πίνακας 6. Η κλίση του επιθέτου «ψηφιακός» όπως παράγεται από το κατηγορήμα *mk_gr_adj/5*

5.1.2.3 Κλίση ρημάτων

Τα ρήματα που αναγνωρίζονται στη διατριβή βρίσκονται στον ενεστώτα, στην ενεργητική φωνή και στην οριστική έγκλιση. Η παθητική φωνή ενός ρήματος μπορεί να αναγνωριστεί, αλλά το ρήμα διαχειρίζεται ως ξεχωριστό ρήμα από αυτό της ενεργητικής φωνής. Με τις επεκτάσεις που προτείνονται σε επόμενο κεφάλαιο μπορούν να αναγνωρίζονται ρήματα σε διαφορετικούς χρόνους, φωνές και εγκλίσεις.

Για τη δημιουργία των κλίσεων των ρημάτων στα τρία πρόσωπα, χρησιμοποιούνται τα γεγονότα από τα αρχεία **verbs.pl** και **verb-suffixes.pl**.

Το αμετάβατο ρήμα «τρέχω» έχει κατάληξη –ω και κλίνεται όπως το μεταβατικό ρήμα «παρέχω». Στο αρχείο **verbs.pl** περιλαμβάνονται τα παρακάτω γεγονότα τα οποία ορίζουν ως ρήματα τις λέξεις «διαβάζω» και «παρέχω».

```
gr_verb_lexicon(τρεχ,1,intrans,2).
gr_verb_lexicon('παρεχ',1,trans,2).
```

Στο αρχείο “**verb-suffixes.pl**” υπάρχουν τα παρακάτω γεγονότα τα οποία ορίζουν τις καταλήξεις των ρημάτων «διαβάζω» και «παρέχω» με βάση την κατηγορία στην οποία ανήκουν.

```
gr_verb_suffix(enikos,proto,1,'ω').
gr_verb_suffix(enikos,deytero,1,'εις').
```

```

gr_verb_suffix(enikos, trito, 1, 'ει').
gr_verb_suffix(plithintikos, proto, 1, 'ουμε').
gr_verb_suffix(plithintikos, deytero, 1, 'ετε').
gr_verb_suffix(plithintikos, trito, 1, 'ουν').

```

Με βάση τα προηγούμενα ο παρακάτω κώδικας της Prolog, αναγνωρίζει τη ρίζα, τον αριθμό, το πρόσωπο, και τη μεταβατικότητα ενός ομαλού ρήματος.

```

gr_verb2 (Riza, Arithmos, Prosopo, Trans, Tropos) -->
    [Word], {
        noMark (Word, NoMarkWord) ,
        gr_verb_lexicon (Riza, Tropos, Trans, Tonos) ,
        sub_atom (NoMarkWord, Before, Length, After, Riza) , Before=0,
        gr_verb_suffix (Arithmos, Prosopo, Tropos, Kataliksi) ,
        atom_concat (Riza, Kataliksi, NoMarkWord) ,
        gr_verb_suffix (Tropos, enikos, proto, KataliksiEnikProto) ,
        mark_rule (verb, Riza, _, KataliksiEnikProto, Tropos, Tonos,
        Prosopo, Arithmos, MarkPos) ,
        atom_concat (Riza, Kataliksi, Word1) ,
        mark (Word1, MarkPos, Word) , !
    } .

```

Η έξοδος από το κατηγορήμα με την παρακάτω κλήση

```

gr_verb2 (Riza, Arithmos, Prosopo, Trans, Tropos, ['παρέχουν'], [])

```

θα επιστρέψει:

```

Riza = παρεχ,
Arithmos = plithintikos,
Prosopo = trito,
Trans = trans,
Tropos = 1 .

```

Όταν σε μια πρόταση εντοπιστεί η λέξη «παρέχουν» και κληθεί το κατηγορήμα **gr_verb2/9**, η εφαρμογή θα αναγνωρίσει ότι πρόκειται για ρήμα στο πρώτο πρόσωπο πληθυντικού με ρίζα «παρεχ». Μετά την ταυτοποίησή της η λέξη «παρέχουν» συνοδεύεται από τις πληροφορίες

Θέμα: παρεχ
Κατάληξη: -ουν
Αριθμός: Πληθυντικός
Πρόσωπο: Τρίτο

Κωδικός κατάληξης: 1
Μέρος του Λόγου: Ρήμα

Οι πληροφορίες αυτές περιλαμβάνονται κωδικοποιημένες στη λίστα
[verb, παρεχ, trans, plithintikos, trito, 1]
η οποία συνοδεύει τη λέξη ως *γραμματική της λέξης*.

Για παράδειγμα για το ρήμα «διαθέτω» δημιουργούνται οι κλίσεις σε όλα τα πρόσωπα και τους αριθμούς, συνδυάζοντας το θέμα με την κατάληξη όπως φαίνεται στον *Πίνακα 7*.

Για την παραγωγή της κλίσης ενός ρήματος σε συγκεκριμένο αριθμό και πρόσωπο χρησιμοποιείται το κατηγορήμα

mk_gr_verb/5

mk_gr_verb (Riza,Arithmos,Prosopo,Trans, Word).

το οποίο συνδυάζοντας το θέμα με τις αντίστοιχες καταλήξεις, παράγει τις κλίσεις του ρήματος για δεδομένο αριθμό και πρόσωπο. Καλώντας το με ρίζα «διαθετ», και τους συνδυασμούς των τιμών για Arithmos=enikos, plithintikos και Prosopo=proto, deytero, trito, δημιουργείται ο *Πίνακας 7* με τις κλίσεις του ρήματος διαθέτω.

Πρόσωπο / Χρόνος	Ενικός	Πληθυντικός
Πρώτο	διαθετ -ω	διαθετ -ουμε
Δεύτερο	διαθετ -εις	διαθετ -ετε
Τρίτο	διαθετ -ει	διαθετ -ουν

Πίνακας 7. Η κλίση του ρήματος «διαθέτω» όπως παράγεται από το κατηγορήμα mk_gr_verb/5

Οι λέξεις που δημιουργούνται από το συνδυασμό του θέματος και της κατάληξης δεν είναι τονισμένες. Οι κανόνες τονισμού εφαρμόζονται στο επόμενο στάδιο. Κάθε μορφή της λέξης που παράχθηκε, συνοδεύεται από τις πληροφορίες της γραμματικής βάσει της οποίας προέκυψε (γραμματική της λέξης).

Για το ανώμαλο ρήμα «είμαι» στο αρχείο **verbs_irregular.pl** υπάρχουν τα παρακάτω γεγονότα.

```
gr_iverb_lexicon('ειμ', 'ειν', enikos, proto, 10, trans) .  
gr_iverb_lexicon('εισ', 'ειν', enikos, deytero, 10, trans) .  
gr_iverb_lexicon('ειν', 'ειν', enikos, trito, 10, trans) .  
gr_iverb_lexicon('ειμ', 'ειν', plithintikos, proto, 10, trans) .  
gr_iverb_lexicon('εισ', 'ειν', plithintikos, deytero, 10, trans) .  
gr_iverb_lexicon('ειν', 'ειν', plithintikos, trito, 10, trans) .
```

και στο αρχείο **verb-suffixes.pl** υπάρχουν τα παρακάτω γεγονότα τα οποία ορίζουν τις καταλήξεις της κατηγορίας 10 στην οποία ανήκει το ρήμα «είμαι».

```
gr_verb_suffix(enikos, proto, 10, 'αι') .  
gr_verb_suffix(enikos, deytero, 10, 'αι') .
```

```
gr_verb_suffix(enikos, trito, 10, 'αι').
gr_verb_suffix(plithintikos, proto, 10, 'αστε').
gr_verb_suffix(plithintikos, deyttero, 10, 'στε').
gr_verb_suffix(plithintikos, trito, 10, 'αι').
```

Για την παρακάτω κλήση

```
gr_verb2(Riza, Arithmos, Prosopo, Trans, Tropos, [είσαι], []).
```

Η έξοδος από το κατηγορήμα **gr_verb2/7** θα επιστρέψει:

```
Riza = ειν,
Arithmos = enikos,
Prosopo = deyttero,
Trans = trans,
Tropos = 10.
```

5.1.2.4 Κλίση άρθρων

Τα άρθρα είναι καταχωρημένα στον πίνακα **articles.pl** σε όλα τα γένη, τους αριθμούς και τις πτώσεις και δε χρησιμοποιείται αρχείο με καταλήξεις.

5.1.3 Συλλαβισμός λέξεων

Ο τονισμός των λέξεων απαιτεί τη δυνατότητα εντοπισμού της λήγουσας, παραλήγουσας και προπαραλήγουσας κάθε λέξης. Για τη λειτουργία αυτή πρέπει να μετρηθούν οι συλλαβές της λέξης ξεκινώντας από το τέλος. Για την απαρίθμηση των συλλαβών δημιουργήθηκε ένα κατηγορήμα το οποίο χωρίζει τις λέξεις σε συλλαβές αναγνωρίζοντας τα σύμφωνα, τα φωνήεντα, τα δίψηφα φωνήεντα (αι, ει, οι, ου, υι), τα δίψηφα σύμφωνα (μπ, ντ, γκ, τσ, τζ) και τους καταχρηστικούς δίφθογγους (αυ και ευ). Το κατηγορήμα

spellCount(StringIn, Count)

δέχεται μία λέξη ως συμβολοσειρά (StringIn) και υπολογίζει τον αριθμό των συλλαβών της λέξης. Π.χ.

```
spellCount('ανταποδίδω', Count).
```

```
Count = 5 .
```

Στον Πίνακα 8 φαίνεται ο αριθμός των συλλαβών που υπολογίζει για κάθε λέξη το κατηγορήμα **spellCount/2**.

Λέξη (StringIn)	Αριθμός Συλλαβών (Count)
άμμος	2
τζιτζικας	3
μπαούλο	3
θειάφι	3
σύμπτωμα	3
εκκλησία	4
ευχαριστώ	4

Πίνακας 8. Ο αριθμός συλλαβών για κάθε λέξη που υπολογίζεται από το κατηγορημα *spellCount/2*

Για τον τονισμό των λέξεων έχει δημιουργηθεί το κατηγορημα

mark(Word, MarkPos, MarkedWord)

το οποίο δέχεται μια λέξη χωρίς τόνους ως συμβολοσειρά (όρισμα Word) και τον αριθμό της συλλαβής που πρόκειται να τονιστεί (όρισμα MarkPos) μετρώντας από το τέλος (1 για τη λήγουσα, 2 για την παραλήγουσα και 3 για την προπαραλήγουσα) και σχηματίζει την τονισμένη λέξη (MarkedWord).

Το κατηγορημα **mark/3** για το αλφαριθμητικό 'ανταποδιδω' και MarkPos=2 επιστρέφει τη λέξη 'ανταποδίδω' τονισμένο στην παραλήγουσα.

```
mark('ανταποδιδω', 2, MarkedWord) .
```

```
MarkedWord = ανταποδίδω .
```

Όμοια για διαφορετικούς συνδυασμούς Word και MarkPos, παράγεται το αντίστοιχο MarkedWord:

```
Word='ευχαριστω', MarkPos=1 → MarkedWord='ευχαριστώ'
```

```
Word='αμμος', MarkPos=2 → MarkedWord='άμμος'
```

```
Word='μπαουλο', MarkPos=2 → MarkedWord='μπαούλο'
```

```
Word='συμπτωμα', MarkPos=3 → MarkedWord='σύμπτωμα'
```

Η πληροφορία για τη συλλαβή στην οποία τονίζεται κάθε λέξη, περιέχεται στα λεξικά των ουσιαστικών, επιθέτων και ρημάτων. Αν ο τόνος μετακινηθεί σε άλλη συλλαβή από αυτήν που είχε στην ονομαστική ή στο πρώτο πρόσωπο του ενικού κατά την κλίση της λέξης, η νέα θέση υπολογίζεται με βάση τους κανόνες τονισμού όπως περιγράφεται στην επόμενη ενότητα.

5.1.4 Τονισμός λέξεων

Οι λέξεις (ουσιαστικά, επίθετα και ρήματα) που παράγονται από το συνδυασμό θέματος και κατάληξης σε διαφορετικές κλίσεις τονίζονται σύμφωνα με τους κανόνες τονισμού της γραμματικής. Κάθε λέξη που προκύπτει από κλίση ουσιαστικού και επιθέτου συνοδεύεται από την πληροφορία του γένους, αριθμού, πτώσης, κατάληξης και αν το ουσιαστικό/επίθετο είναι οξύτονο, παροξύτονο ή προπαροξύτονο. Όμοια κάθε λέξη που προκύπτει από κλίση ρήματος συνοδεύεται από την πληροφορία του προσώπου, του αριθμού, της κατάληξης και αν το ρήμα είναι οξύτονο, παροξύτονο ή προπαροξύτονο. Με βάση αυτές τις πληροφορίες εφαρμόζονται οι γραμματικοί κανόνες τονισμού των

λέξεων και παράγεται η τονισμένη λέξη. Οι κανόνες τονισμού που χρησιμοποιούνται περιέχονται στο αρχείο **mark_rules.pl** της Prolog ως γεγονότα και έχουν τη μορφή

mark_rule(ΜΤΛ, Ρίζα, Γένος, Κατάληξη, Τρόπος, Τόνος, Πτώση_Πρόσωπο, Αριθμός, Θέση_Τόνου).

όπου

- «ΜΤΛ» είναι το μέρος του λόγου στο οποίο ανήκει η λέξη με τιμές noun και verb.
- «Ρίζα» είναι είναι το θέμα της λέξης.
- «Γένος» είναι το γένος του ουσιαστικού ή του επιθέτου. Αν πρόκειται για ρήμα η τιμή αγνοείται.
- «Κατάληξη» είναι η κατάληξη της λέξης.
- «Τρόπος» είναι ο κωδικός κατάληξης της λέξης.
- «Τόνος» είναι ένας αριθμός με τιμές 1, 2 και 3 που δείχνει αν η λέξη τονίζεται στη λήγουσα, την παραλήγουσα, ή την προπαραλήγουσα στην ονομαστική του ενικού για τα ουσιαστικά και τα επίθετα ή στο πρώτο πρόσωπο του ενικού για τα ρήματα.
- «Πτώση_Πρόσωπο» είναι η πτώση του ουσιαστικού/επιθέτου ή το πρόσωπο του ρήματος στα οποία βρίσκεται η λέξη που πρόκειται να τονιστεί.
- «Αριθμός» είναι ο αριθμός στον οποίο βρίσκεται η λέξη που πρόκειται να τονιστεί.
- «Θέση_Τόνου» είναι ένας αριθμός με τιμές 1, 2 και 3 που δείχνει σε ποια συλλαβή (λήγουσα, παραλήγουσα, προπαραλήγουσα) πρόκειται να τονιστεί η λέξη στο συγκεκριμένο αριθμό και πτώση ή πρόσωπο. Η «Θέση_Τόνου» υπολογίζεται σύμφωνα με τις προηγούμενες παραμέτρους και τους κανόνες τονισμού.

Οι κανόνες τονισμού που χρησιμοποιούνται, συνοψίζονται παρακάτω.

Ουσιαστικά/Επίθετα

1. Τα παροξύτονα σε -ας τονίζονται στη γενική πληθυντικού στην παραλήγουσα (π.χ αγώνας-αγώνες)
2. Τα ανισοσύλλαβα αρσενικά σε -ας τονίζονται στον πληθυντικό στην παραλήγουσα (π.χ σφουγγαράς-σφουγγαράδες).
3. Τα παροξύτονα σε -ης τονίζονται στη γενική πληθυντικού στην λήγουσα (π.χ επιβάτης-επιβατών).
4. Τα προπαροξύτονα αρσενικά σε -ος τονίζονται στη γενική του ενικού στην παραλήγουσα (π.χ. άνθρωπος-ανθρώπου).
5. Τα προπαροξύτονα θηλυκά σε -ος τονίζονται στη γενική του ενικού και πληθυντικού στην παραλήγουσα (π.χ είσοδος-εισόδου, εισόδων).
6. Τα προπαροξύτονα ουδέτερα σε -ος τονίζονται στη γενική του ενικού και στην ονομαστική και αιτιατική πληθυντικού στην παραλήγουσα (π.χ μέγεθος-μεγέθους, μεγέθη).
7. Τα ουδέτερα σε -ος τονίζονται στη γενική του πληθυντικού στη λήγουσα (π.χ μέρος-μερών).
8. Τα παροξύτονα ουδέτερα σε -ο τονίζονται στη γενική του ενικού και πληθυντικού στην παραλήγουσα (έγγραφο-εγγράφου, εγγράφων).

9. Τα ουδέτερα σε -ιο τονίζονται στη γενική του ενικού και πληθυντικού στην παραλήγουσα (βιβλίο-βιβλίου, βιβλίων και τετράδιο-τετραδίου, τετραδίων).
10. Τα παροξύτονα ανισοσύλλαβα ουδέτερα σε -α και -ας τονίζονται στη γενική του ενικού και στην ονομαστική και αιτιατική πληθυντικού στην προπαραλήγουσα (κύμα-κύματος, κύματα και κρέας-κρέατος, κρέατα).
11. Αν δεν έχει ικανοποιηθεί κανένας από τους προηγούμενους κανόνες, ο τόνος δε θα μετακινηθεί κατά την κλίση (θα μείνει στη συλλαβή της ονομαστικής ή πρώτου προσώπου ενικού).

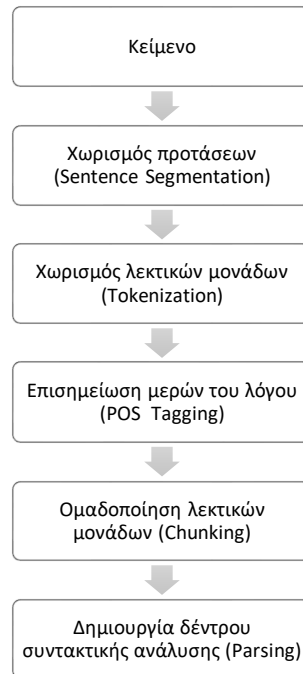
Ρήματα

1. Τα οξύτονα ρήματα που τελειώνουν σε -ω τονίζονται στο πρώτο και δεύτερο πρόσωπο του πληθυντικού στην παραλήγουσα (περνώ-περνάμε, περνάτε).
2. Τα παροξύτονα ρήματα που τελειώνουν σε -ω τονίζονται στο πρώτο και δεύτερο πρόσωπο του πληθυντικού στην προπαραλήγουσα (δένω-δένουμε, δένετε).
3. Αν δεν έχει ικανοποιηθεί κανένας από τους προηγούμενους κανόνες, η θέση του τόνου θα παραμείνει αυτή που έχει δηλωθεί στο λεξικό του ρήματος (θα μείνει στη συλλαβή του πρώτου προσώπου ενικού).

Μετά την παραγωγή κάθε μορφής της λέξης με τους συνδυασμούς θέματος - κατάληξης, καλείται το κατηγορημα **mark_rules/9** με παραμέτρους τα στοιχεία της γραμματικής της λέξης που αναφέρθηκαν και εφαρμόζοντας τους κανόνες τονισμού επιστρέφει τη θέση του τόνου της λέξης. Ακολούθως καλείται το κατηγορημα **mark/3** που δέχεται τη λέξη χωρίς τόνο και τον αριθμό που δηλώνει τη θέση του τόνου και επιστρέφει τη λέξη τονισμένη.

Οι κανόνες τονισμού μπορούν να εμπλουτιστούν περισσότερο προσθέτοντας νέους που αφορούν σε εξαιρέσεις, ανώμαλα ουσιαστικά, επίθετα και ρήματα, χρόνους και φωνές ρημάτων κ.α. με την προσθήκη νέων κατηγορημάτων στο αρχείο **mark_rules.pl**.

5.1.5 Διαδικασία δημιουργίας δέντρου συντακτικής ανάλυσης



Σχήμα 11. Τα στάδια για τη δημιουργία του δέντρου συντακτικής ανάλυσης

Το κείμενο που εισάγεται από το χρήστη επεξεργάζεται με μία σειρά διαδικασιών όπως φαίνεται στο Σχήμα 11, ώστε στο τέλος να δημιουργηθεί μία δομή που θα αναπαριστά το δέντρο συντακτικής ανάλυσης (parse tree). Αρχικά το κείμενο χωρίζεται σε προτάσεις (Sentence Segmentation) κάθε μία από τις οποίες επεξεργάζεται χωριστά. Στη συνέχεια κάθε πρόταση χωρίζεται (Tokenization) σε λεκτικές μονάδες (tokens) οι οποίες είναι λέξεις και αριθμοί. Κάθε λέξη μετά από μορφολογική ανάλυση εντάσσεται σε γραμματική κατηγορία και κλιτική τάξη (Tagged POS). Οι λεκτικές μονάδες εντάσσονται σε συντακτικές μονάδες όπως οι Ονοματικές και Ρηματικές Φράσεις (Chunking) και καθορίζεται ο ρόλος τους σε αυτές. Οι συντακτικές μονάδες συσχετίζονται μεταξύ τους (parsing) και δημιουργείται το δέντρο συντακτικής ανάλυσης (parse tree). Για τη δημιουργία του δέντρου συντακτικής ανάλυσης, εκτελούνται διαδοχικά οι διαδικασίες όπως περιγράφονται στη συνέχεια.

5.1.5.1 Χωρισμός προτάσεων (Sentence Segmentation)

Το κείμενο που εισάγεται από το χρήστη, χωρίζεται σε προτάσεις. Ο εντοπισμός των ορίων κάθε πρότασης γίνεται με αναζήτηση της τελείας («.»). Κάθε πρόταση προστίθεται σε μία λίστα, από την οποία θα ανακτηθεί στη συνέχεια και θα επεξεργαστεί.

5.1.5.2 Χωρισμός λεκτικών μονάδων (Tokenization)

Κάθε πρόταση χωρίζεται σε λεκτικές μονάδες (tokens). Κάθε λεκτική μονάδα συγκρίνεται με τις λέξεις του λεξικού σε όλες τις κλιτικές τάξεις (εάν υπάρχουν) και εφόσον συμφωνεί με κάποια από αυτές θεωρείται γνωστή λέξη. Ο αλγόριθμος της ταυτοποίησης των λέξεων περιγράφεται στο επόμενο βήμα. Οι αριθμοί θεωρούνται γνωστές λέξεις. Αν η λέξη δεν είναι γνωστή θεωρείται άγνωστη και ενημερώνεται ο χρήστης με σχετικό μήνυμα. Οι λόγοι για τους οποίους μία λέξη θεωρείται άγνωστη είναι είτε γιατί δεν υπάρχει στο λεξικό, είτε γιατί περιέχει ορθογραφικά λάθη. Μετά από την προσθήκη μιας άγνωστης λέξης στο λεξικό ή τη διόρθωση των ορθογραφικών λαθών εφόσον υπάρχουν, η διαδικασία μπορεί να επαναληφθεί και να προχωρήσει στα επόμενα βήματα.

5.1.5.3 Επισημείωση μερών του λόγου (POS Tagging)

Κάθε λέξη ελέγχεται με σκοπό τον εντοπισμό του μέρους του λόγου στο οποίο ανήκει και στοιχείων για την κλίση της εφόσον υπάρχουν. Στα *άρθρα*, τα *ουσιαστικά*, τα *επίθετα* και στις *αντωνυμίες* καθορίζονται το *γένος*, ο *αριθμός* και η *πτώση* ενώ στα *ρήματα* καθορίζεται ο *αριθμός* και το *πρόσωπο*. Η διαδικασία είναι παρόμοια με αυτή του προηγούμενου βήματος, δηλαδή η μορφή κάθε λέξης παράγεται σε διαφορετικές κλίσεις και συγκρίνεται με τα δεδομένα από το λεξικό, χρησιμοποιώντας γραμματικούς κανόνες. Σε περίπτωση ταύτησης θεωρείται γνωστή η γραμματική κατηγορία και η κλιτική τάξη της λέξης.

Για τη δημιουργία της λέξης σε διαφορετικές κλίσεις, χρησιμοποιούνται οι πίνακες των λεξικών που περιέχουν τις ρίζες των λέξεων σε συνδυασμό με τις αντίστοιχες καταλήξεις. Έτσι παράγονται τα ουσιαστικά σε όλες τις διαφορετικές πτώσεις και αριθμούς, τα επίθετα σε όλα τα γένη, πτώσεις και αριθμούς, τα ρήματα σε όλα τα πρόσωπα και αριθμούς κλπ. Οι λέξεις που δημιουργούνται στις διάφορες κλίσεις συγκρίνονται με τη λέξη που ελέγχεται κάθε φορά και εφόσον ταυτίζονται θεωρείται ότι εντοπίστηκε το μέρος του λόγου και η κλιτική τάξη της ελεγχόμενης λέξης. Η σύγκριση γίνεται σε τρία στάδια. Στο πρώτο στάδιο ελέγχεται αν ταυτίζεται το θέμα της λέξης του λεξικού με τους αρχικούς χαρακτήρες της ελεγχόμενης λέξης του κειμένου. Στο δεύτερο στάδιο ελέγχεται αν ταυτίζεται η λέξη που παρήχθη από τις κλίσεις με την ελεγχόμενη λέξη του κειμένου αγνοώντας τους τόνους. Αν υπάρχει ταύτιση τότε είναι πλέον γνωστό το μέρος του λόγου και τα αντίστοιχα στοιχεία κλίσης της λέξης (πτώση, πρόσωπο, αριθμός, γένος κλπ). Στη συνέχεια με βάση τους γραμματικούς κανόνες με τους οποίους παρήχθη η λέξη εφαρμόζονται οι κανόνες τονισμού σ' αυτήν (π.χ ανισοσύλλαβο αρσενικό ουσιαστικό σε -ος στη γενική πληθυντικού) και παράγεται η ίδια λέξη τονισμένη. Αυτή η λέξη συγκρίνεται με την ελεγχόμενη τονισμένη λέξη και εφόσον ταυτίζονται θεωρείται ότι η λέξη έχει ταυτοποιηθεί. Η σύγκριση γίνεται σε τρία στάδια ώστε να μην παράγονται διαφορετικές κλίσεις και γίνονται συγκρίσεις σε λέξεις που έχουν διαφορετικά θέματα και επιπλέον οι κανόνες τονισμού να μην εφαρμόζονται σε κάθε λέξη που δημιουργείται από το λεξικό, αλλά μόνο εφόσον οι συγκρινόμενες λέξεις είναι όμοιες. Έτσι ο αλγόριθμος εντοπισμού ολοκληρώνεται συντομότερα αποφεύγοντας άσκοπους ελέγχους.

Τα βήματα που ακολουθούνται για την ταυτοποίηση κάθε λέξης είναι τα εξής:

1. Σύγκριση το θέμα της λέξης του λεξικού με την αρχή της λέξης του κειμένου (σύγκριση χωρίς τους τόνους).

2. Εφόσον ταυτίζονται δημιουργήσε τις κλίσεις της λέξης του λεξικού και σύγκρινέ τες με τη λέξη του κειμένου (σύγκριση χωρίς τους τόνους).
3. Εφόσον ταυτίζονται εφάρμοσε τους κανόνες τονισμού στη λέξη του λεξικού και συγκρινέ την με τη λέξη του κειμένου.
4. Εφόσον ταυτίζονται η λέξη έχει ταυτοποιηθεί.

Ο αλγόριθμος για την ταυτοποίηση των λέξεων φαίνεται παρακάτω:

Δημιούργησε τη λίστα λέξεων SentenceWords από την πρόταση S

Για κάθε λέξη Word της λίστας SentenceWords **κάνε**

ΤαυτοποίησηΛέξης(*Word, GrammarWordTuple*);

Τέλος Για κάθε

Διαδικασία ΤαυτοποίησηΛέξης(*Word, GrammarWordTuple*)

Αρχή_διαδ

flag_found=FALSE ;

Έστω το σύνολο Lex_Suf = {(noun_lexicon, noun_suffixes) , (verb_lexicon, verb_ suffixes) (adjective_lexicon, adjective_ suffixes) }

Ενώ Lex_Suf ≠ {} **και** flag_found=FALSE **κάνε**

Έστω (X_lexicon, X_suffixes) ∈ Lex_Suf ;

Ενώ X_lexicon ≠ {} **και** flag_found=FALSE **κάνε**

Έστω εγγραφή R_lex ∈ X_lexicon;

Έστω root η τιμή του πεδίου του 'θέματος' της εγγραφής R_lex;

Εάν το root ταιριάζει με τους αντίστοιχους πρώτους χαρακτήρες του SearchWord **τότε**

Έστω suff_id η τιμή του πεδίου 'κωδικός_καταληξης';

Έστω το σύνολο R_all_suff = {R_suff: R_suff ∈ X_suffixes με 'Κωδικό_Κατάληξης' suff_id};

Ενώ R_all_suff ≠ {} **και** flag_found=FALSE **κάνε**

Έστω εγγραφή R_s ∈ R_all_suff;

Έστω suffix η τιμή του πεδίου 'Κατάληξη';

Δημιούργησε τη συμβολοσειρά Word = root + suffix

Αν Word = SearchWord **τότε**

flag_found=TRUE

GrammarWordTuple := get_grammar_word(R_s);

Τέλος αν

R_all_suff := R_all_suff - {R_s};

Τέλος_ενώ

Τέλος_εάν

Lex_Suf := Lex_Suf - {(X_lexicon, X_suffixes)};

Τέλος ενώ

Τέλος_διαδ

Σε περίπτωση που σε κάποιο από τα επόμενα βήματα, με τη συγκεκριμένη κλιτική τάξη δεν ικανοποιούνται οι γραμματικοί και συντακτικοί κανόνες ώστε να δημιουργηθεί τελικά το δέντρο συντακτικής ανάλυσης, η διαδικασία της επεξεργασίας θα επιστρέψει σε αυτό το βήμα και θα αναζητήσει μια άλλη κλιτική τάξη. Αν για παράδειγμα ένα ουδέτερο ουσιαστικό έχει την ίδια μορφή στην ονομαστική και την αιτιατική του ενικού, η διαδικασία θα προχωρήσει στα επόμενα βήματα αρχικά, θεωρώντας ότι η λέξη βρίσκεται σε ονομαστική ενικού και εφόσον η ονομαστική δεν ταιριάζει γραμματικά ή συντακτικά στη δομή της πρότασης, η διαδικασία θα επιστρέψει σε αυτό το βήμα και θα συνεχίσει στα επόμενα χρησιμοποιώντας την εκδοχή της αιτιατικής. Τα δεδομένα της γραμματικής κατηγορίας, της κλιτικής τάξης και οποιαδήποτε άλλη πληροφορία για τη μορφή και το είδος της λέξης, αποθηκεύεται σε μια λίστα της Prolog όπως φαίνεται παρακάτω η οποία χαρακτηρίζει γραμματικά και συνοδεύει τη λέξη. Η λίστα αυτή θα αναφέρεται με τον όρο «Γραμματική της λέξης» και το όρισμα *WordGrammar*. Η «Γραμματική της λέξης» έχει τη μορφή:

[Μέρος_του_λόγου,Ρίζα_Λέξης|Κλιτική_Τάξη]

Η Prolog λίστα «Κλιτική_Τάξη» για **άρθρα, ουσιαστικά, επίθετα** και **αντωνυμίες** έχει τη μορφή:

[Γένος, Αριθμός, Πτώση, Κωδικός_Κατάληξης]

Η Prolog λίστα «Κλιτική_Τάξη» για **ρήματα** έχει τη μορφή:

[Μεταβατικότητα, Αριθμός, Πρόσωπο, Κωδικός_Κατάληξης]

Μετά από τη δημιουργία του δέντρου συντακτικής ανάλυσης, η «Γραμματική της λέξης» αποθηκεύεται μαζί με τη λέξη και χρησιμοποιείται από τις επόμενες λειτουργίες της εφαρμογής.

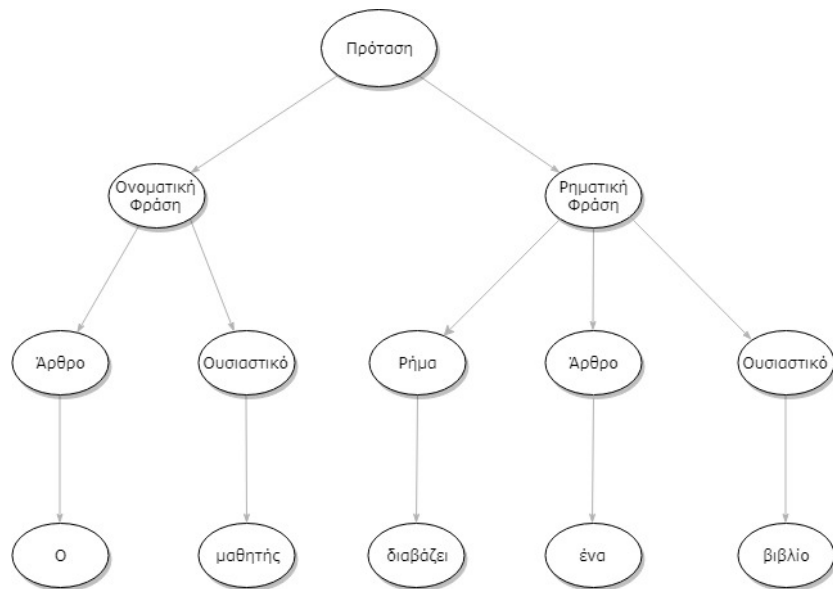
5.1.5.4 Ομαδοποίηση λεκτικών μονάδων (Chunking)

Οι λεκτικές μονάδες μετά την επισημείωσή τους, ενώνονται σε ομάδες οι οποίες αποτελούν συντακτικές μονάδες της πρότασης όπως οι Ονομαστικές και οι Ρηματικές Φράσεις. Με βάση τους

συντακτικούς και γραμματικούς κανόνες, καθορίζεται ο ρόλος των λέξεων και ο τρόπος με τον οποίο οι λέξεις συνδέονται μεταξύ τους. Για παράδειγμα ένα επίθετο που προηγείται ενός ουσιαστικού είναι επιθετικός προσδιορισμός του ουσιαστικού εφόσον βρίσκεται στο ίδιο γένος, πτώση και αριθμό με το ουσιαστικό. Δηλαδή εντοπίζονται πρώτα το γένος, η πτώση και ο αριθμός του ουσιαστικού και στη συνέχεια ελέγχεται αν η μορφή του επιθέτου που προηγείται έχει την ίδια κλιτική τάξη με το ουσιαστικό. Εφόσον συμφωνούν τότε το επίθετο συνδέεται ως επιθετικός προσδιορισμός με το ουσιαστικό και αποτελούν την ονομαστική φράση.

5.1.5.5 Δημιουργία δέντρου συντακτικής ανάλυσης (Parsing)

Μετά το έλεγχο της συντακτικής ορθότητας των λέξεων ακολουθεί η συντακτική ορθότητα της πρότασης. Η συντακτική ορθότητα των προτάσεων εξετάζεται σε σχέση με τη γραμματική της γλώσσας. Ένας συντακτικός αναλυτής ή αναλυτής αναγνωρίζει προτάσεις που ανήκουν στη γραμματική της γλώσσας. Ο συντακτικός αναλυτής εφαρμόζει τους συντακτικούς κανόνες στις ομάδες των λέξεων που αναφέρθηκαν στο προηγούμενο βήμα, και εφόσον η δομή της πρότασης ταιριάζει με αυτούς, παράγεται το δέντρο συντακτικής ανάλυσης (parse tree) στο οποίο εμφανίζονται οι λέξεις της πρότασης και οι σχέσεις εξάρτησης μεταξύ τους. Οι συντακτικοί κανόνες της Ελληνικής γραμματικής παριστάνονται στην Prolog με DCG κανόνες. Στο Σχήμα 12 απεικονίζεται το συντακτικό δέντρο της πρότασης «Ο μαθητής διαβάζει ένα βιβλίο».



Σχήμα 12. Το δέντρο σύνταξης της πρότασης «Ο μαθητής διαβάζει ένα βιβλίο»

Με βάση τη γραμματική που ορίστηκε μπορεί να δημιουργηθεί το δέντρο σύνταξης (parse tree) για κάθε πρόταση που εισάγεται στην εφαρμογή. Επίσης εξετάζεται αν συμφωνεί ο αριθμός του

ουσιαστικού και του ρήματος καθώς και η πτώση του άρθρου και του ουσιαστικού, με συνέπεια προτάσεις όπως:

- Οι μαθητής διαβάζει ένα βιβλίο
- Ο μαθητής διαβάζουν ένα βιβλίο

να μη γίνονται αποδεκτές από τη γραμματική.

5.1.6 Κανόνες γραμματικής της ελληνικής γλώσσας

Η ελληνική γλώσσα έχει πολύπλοκους κανόνες που διέπουν τη δημιουργία φράσεων. Στην διατριβή χρησιμοποιείται ένα μικρό σύνολο βασικών κανόνων γραμματικής της Ελληνικής γλώσσας. Η τυπική γραμματική της γλώσσας ορίζεται από την τετράδα (N, Σ, Ρ, S) όπου:

- το σύνολο από μη-τερματικά σύμβολα **N** είναι λέξεις ή φράσεις μέσα σε εισαγωγικά οι οποίες ξεκινάνε με κεφαλαίο γράμμα όπως <Πρόταση>, <Ονοματικό Μέρος> κτλ. N= {<Πρόταση>, <Ρηματικό μέρος>, <Ονοματικό μέρος>, <Ουσιαστικό>, <Ρήμα>, <Επίθετο>, <Άρθρο>, <Σύνδεσμος>, ..}
- το σύνολο από τερματικά σύμβολα **Σ** με τις λέξεις που περιέχονται ή παράγονται από το λεξικό της εφαρμογής και οι οποίες ξεκινάνε με πεζό γράμμα του Ελληνικού αλφάβητου εκτός από τα κύρια ονόματα που μπορούν να ξεκινούν με κεφαλαίο γράμμα του Ελληνικού αλφάβητου. Για παράδειγμα φαγητό, Ηράκλειο, κτλ.
- το αρχικό σύμβολο **S** είναι S = <Πρόταση>
- το σύνολο P αποτελείται από τους κανόνες παραγωγής σε μορφή BNF που περιγράφονται στη συνέχεια.

Οι κανόνες BNF της γραμματικής της εφαρμογής διατυπώνονται παρακάτω:

Πρόταση ::= <Ονοματικό_Μέρος> <Ρηματικό_Μέρος> | <Πρόταση> <Ρηματικό_Μέρος>.

<Ρηματικό_Μέρος> ::=
<Αντωνυμία> <Ρήμα> [<Αντικείμενο>] [<Επίρρημα>] | <Ρήμα> [<Αντικείμενο>] [<Επίρρημα>] |
<Ρήμα> <Σύνδεσμος> <Ρήμα>

Ονοματικό_Μέρος ::=
<Υποκείμενο> | <Υποκείμενο> <Εμπρόθετος_Προσδιορισμός>

Αντικείμενο ::=
<Εμπρόθετος_Προσδιορισμός> | <Επιθετικός_Προσδιορισμός> |
[<Άρθρο>] [<Επιθετικός_Προσδιορισμός> | <Επιθετικός_Προσδιορισμός>] <Ουσιαστικό> |
<Άρθρο> <Ουσιαστικό> [<Άρθρο>] <Ετερόπτωτος_Ονοματικός_Προσδιορισμός_Γενική> |
<Άρθρο> [<Επιθετικός_Προσδιορισμός>] <Ουσιαστικό> [<Ουσιαστικό>] |
<Εμπρόθετος_Προσδιορισμός >

Υποκείμενο ::=
[<Άρθρο>] [<Επιθετικός_Προσδιορισμός>] <Ουσιαστικό> |

<Άρθρο> <Ουσιαστικό> [<Άρθρο>] <Ετερόπτωτος_Ονοματικός_Προσδιορισμός_Γενική>|
 [<Άρθρο>] [<Επιθετικός Προσδιορισμός>] <Ουσιαστικό> [<Άρθρο>]
 <Ετερόπτωτος_Ονοματικός_Προσδιορισμός_Γενική>
 [<Ετερόπτωτος_Ονοματικός_Προσδιορισμός_Γενική>]
 [<Άρθρο>] <Ουσιαστικό> <Σύνδεσμος> <Ουσιαστικό> |
 [<Άρθρο>] <Ουσιαστικό> <Ουσιαστικό> |
 [<Άρθρο>] <Ουσιαστικό> <Πρόθεση> [<Ρήμα> | < Ουσιαστικό >]

Επιθετικός_Προσδιορισμός ::= <Επίθετο> | <Επίθετο> <Σύνδεσμος> <Επίθετο>

Εμπρόθετος_Προσδιορισμός ::= <Πρόθεση> [<Επιθετικός Προσδιορισμός>] <Ουσιαστικό>

Ετερόπτωτος_Ονοματικός_Προσδιορισμός_Γενική ::=
 [<Άρθρο>] [<Επιθετικός_Προσδιορισμός>] <Ουσιαστικό>

Με βάση αυτούς τους κανόνες BNF αναλύονται οι προτάσεις που εισάγονται στην εφαρμογή και δημιουργούνται οι απαντήσεις που παρέχει προς το χρήστη.

Παράδειγμα 1:

Παρακάτω φαίνονται οι κανόνες BNF που ικανοποιούνται κατά την ανάλυση της φράσης «*Ο Νίκος μελετάει ιστορία εντατικά*»

<Πρόταση> ::= <Ονοματικό_Μέρος> <Ρηματικό_Μέρος>
 <Ονοματικό_Μέρος> ::= <Υποκείμενο>
 <Ρηματικό_Μέρος> ::= <Ρήμα> <Αντικείμενο> <Επίρρημα>
 <Αντικείμενο> ::= <Ουσιαστικό>
 <Υποκείμενο> ::= <Άρθρο> <Ουσιαστικό>
 <Ουσιαστικό> ::= "Νίκος" | "ιστορία"
 <Ρήμα> ::= "μελετάω"
 <Επίρρημα> ::= "εντατικά"
 <Άρθρο> ::= "ο"

Παράδειγμα 2:

Παρακάτω φαίνονται οι κανόνες BNF που ικανοποιούνται κατά την ανάλυση της φράσης «*Η γρίπη είναι μια μεταδοτική ασθένεια των θηλαστικών*»

Πρόταση ::= <Ονοματικό_Μέρος> <Ρηματικό_Μέρος>
 Ονοματικό_Μέρος ::= <Υποκείμενο>
 Ρηματικό_Μέρος ::= <Ρήμα> <Αντικείμενο>
 Αντικείμενο ::= <Επιθετικός Προσδιορισμός> <Ουσιαστικό> <Άρθρο>
 <Ετερόπτωτος_Ονοματικός_Προσδιορισμός_Γενική>
 Υποκείμενο ::= <Άρθρο> <Ουσιαστικό>
 Επιθετικός_Προσδιορισμός ::= <Επίθετο> <Επίθετο>
 Ρήμα ::= "είμαι"
 Επίθετο ::= "ένας" | "μεταδοτικός"
 Ουσιαστικό ::= "γρίπη" | "ασθένεια" | "θηλαστικά"
 Άρθρο ::= "ο"

Όμοια με την εφαρμογή των γραμματικών κανόνων BNF μπορεί να δημιουργηθεί το συντακτικό δέντρο των προτάσεων από τα ακόλουθα παραδείγματα.

- «Ο Γιάννης περνά».
- «Ο Γιάννης περνά τώρα».
- «Ο ουρανός είναι γαλάζιος».
- «Η αδελφή της Μαρίας περνά».
- «Η αδελφή της Μαρίας περνά καθημερινά».
- «Ο μαθητής Νίκος διαβάζει ένα μεγάλο βιβλίο».
- «Ο μαθητής Νίκος διαβάζει ένα μεγάλο βιβλίο γρήγορα».
- «Οι μαθητές Νίκος και Μαρία διαβάζουν ένα μεγάλο βιβλίο γρήγορα».
- «Ο Νίκος φοράει άσπρο καπέλο στο κεφάλι».
- «Το γενετικό υλικό του παραμυξιοίου είναι το μονόκλωνο ριβονουκλεικό οξύ».
- «Η κλινική διάγνωση της ιλαράς απαιτεί ιστορικό πυρετού λίγων ημερών».
- «Το λεμόνι δίνει νόστιμη γεύση στο φαγητό».
- «Η κεντρική τράπεζα καθορίζει το μέγιστο ονομαστικό επιτόκιο της εξάμηνης προθεσμιακής κατάθεσης καθημερινά».
- «Η όμορφη ψηλή γυναίκα του νέου μανάβη της γειτονιάς που περνά τώρα φοράει μοντέρνο πράσινο φόρεμα».
- «Οι μεγαλύτεροι πλανήτες του ηλιακού συστήματος είναι οι πλανήτες Δίας και Κρόνος».
- «Το μέγιστο ηλεκτρικό ρεύμα φόρτισης της μπαταρίας του φωτοβολταϊκού συστήματος που προτείνεται ανεπιφύλακτα αναφέρεται στο συνοδευτικό εγχειρίδιο».
- «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά».

Για την υλοποίηση της λεξικολογικής και γραμματικής ανάλυσης στην Prolog χρησιμοποιείται η Γραμματική Οριστικών Προτάσεων Definite Clause Grammar (DCG). Με τη Γραμματική Οριστικών Προτάσεων η μεταφορά των κανόνων παραγωγής σε κατηγορήματα της Prolog, γίνεται άμεσα. Για παράδειγμα ο κανόνας

<Πρόταση> ::= <Ονοματικό Μέρος> <Ρηματικό μέρος>

γράφεται ως

sentence --> noun_phrase, verb_phrase.

Σ' αυτή τη φάση έχει καθοριστεί πλήρως ο ρόλος κάθε λέξης (ρίζα, γραμματική, συντακτικό) μέσα στην πρόταση ο οποίος χρησιμοποιείται για τον καθορισμό των οντοτήτων στην επόμενη φάση.

5.1.7 Αναπαράσταση της δομής των προτάσεων

Η δομή κάθε πρότασης εκφράζεται με μορφή κατηγορήματος της Prolog. Στις παραμέτρους περιέχονται τα επίθετα, ουσιαστικά, επιρρήματα, μαζί με στοιχεία της γραμματικής τους ή άλλες δομές ονοματικών και ρηματικών φράσεων που συνδέονται με την λέξη αυτή.

Κάθε φορά που μια ομάδα λέξεων ικανοποιεί κάποιον από τους εφαρμοζόμενους κανόνες της γραμματικής DCG, δημιουργείται μια δομή η οποία εκφράζει τις συντακτικές σχέσεις μεταξύ των λέξεων και της γραμματικής τους.

Για κάθε ρήμα της πρότασης δημιουργείται μια ξεχωριστή «Δομή Ρήματος» που περιέχει τη λίστα μίας η περισσότερων «Δομών Ουσιαστικού» οι οποίες συνδέονται συντακτικά με το ρήμα ως υποκείμενο και μια λίστα μίας η περισσότερων «Δομών Ουσιαστικού» οι οποίες συνδέονται συντακτικά με το ρήμα ως αντικείμενο. Δημιουργούνται δηλαδή τόσες «Δομές Ρήματος», όσα και τα ρήματα της πρότασης.

Για παράδειγμα η δομή της πρότασης «*Ο μαθητής διαβάζει ένα βιβλίο*», αναπαρίσταται ως

δ ι α β α ζ ([μ α θ η τ (...), [β ι β λ (...)]], ...) .

Τα αποσιωπητικά αντικαθιστούν παραμέτρους οι οποίες περιγράφονται παρακάτω για λόγους απλότητας.

Τα άρθρα και οι αντωνυμίες δεν περιέχονται στις δομές.

5.1.7.1 Αναπαράσταση δομής «μερών του λόγου» ως όροι της Prolog

5.1.7.1.1 Δομή πρότασης

Η δομή της πρότασης έχει ως κεντρικό κόμβο το ρήμα και αναπαριστά τη σύνδεσή του με το υποκείμενο και το αντικείμενο. Όλες οι προτάσεις αναπαρίστανται από ένα ρήμα που συνδέεται με μία δομή Ονοματικής Φράσης. Η Ονοματική Φράση είναι το υποκείμενο του ρήματος. Αν το ρήμα είναι μεταβατικό, τότε συνδέεται και με μία δεύτερη Ονοματική Φράση, η οποία είναι το αντικείμενο του ρήματος. Η γενική μορφή αναπαράστασης της δομής μιας πρότασης είναι:

Ρ ί ζ α Ρ ή μ α τ ο ς (

[Υποκείμενο, Αντικείμενο],
Επιρρηματικός_Προσδιορισμός,
Λίστα_γραμματικής_ρήματος

)

Η παράμετρος «Υποκείμενο» είναι μια λίστα από ένα ή περισσότερα ουσιαστικά που συνδέονται συντακτικά με το ρήμα και σχηματίζουν το υποκείμενο της πρότασης. Η παράμετρος «Υποκείμενο» έχει τη μορφή:

Υποκείμενο =

[

Δομή_ουσιαστικού 1,

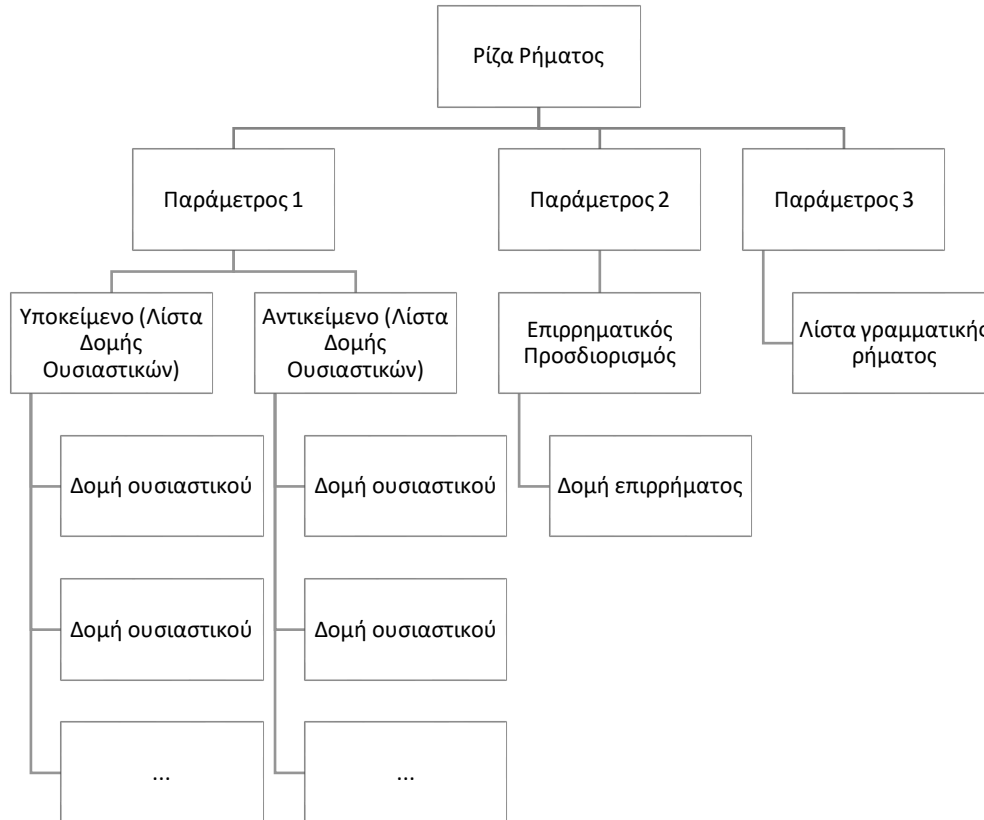
Δομή_ουσιαστικού 2,

..

]

Η μορφή που έχει η «Δομή_ουσιαστικού» περιγράφεται στη συνέχεια.

Αντίστοιχα η παράμετρος «Αντικείμενο» είναι μια λίστα από ένα ή περισσότερα ουσιαστικά που συνδέονται συντακτικά με το ρήμα ως αντικείμενα. Η παράμετρος «Αντικείμενο» ορίζεται μόνο για μεταβατικά ρήματα. Η παράμετρος «Επιρρηματικός_Προσδιορισμός» είναι μία λίστα που περιέχει ως στοιχείο τη «Δομή Επιρρήματος».



Σχήμα 13: Η γενική αναπαράσταση της δομής μιας πρότασης

Στο Σχήμα 13 απεικονίζεται γραφικά ο σχηματισμός της δομής μιας πρότασης και ο τρόπος που συνδέονται τα στοιχεία της δομής μεταξύ τους.

5.1.7.1.2 Δομή ουσιαστικού

Η Δομή Ουσιαστικού έχει ως κεντρικό κόμβο το ουσιαστικό και αναπαριστά τη σύνδεσή του με άλλα ουσιαστικά και επιθετικούς προσδιορισμούς. Η γενική μορφή αναπαράστασης της δομής ενός ουσιαστικού είναι:

ΡίζαΟυσιαστικού (

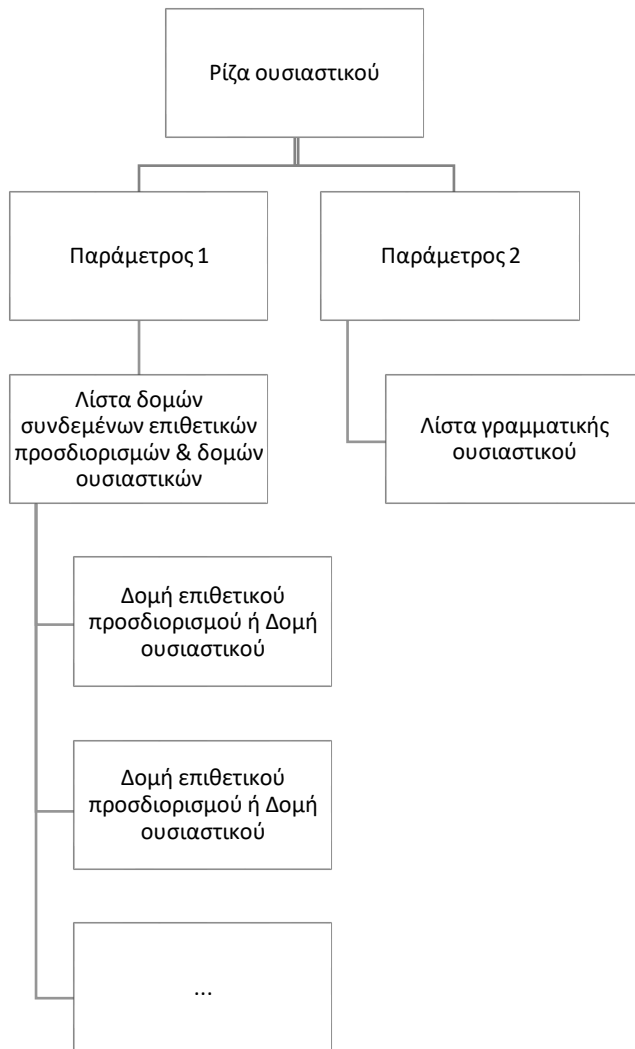
Λίστα_δομών_συνδεδεμένων_σε_ουσιαστικό_λέξεων,

Λίστα_γραμματικής_ουσιαστικού

)

Η «Λίστα_δομών_συνδεδεμένων_σε_ουσιαστικό_λέξεων» είναι μια λίστα που αποτελείται από μία (ή περισσότερες) «Δομή επιθετικού προσδιορισμού» ή/και από μία (ή περισσότερες) «Δομή

ουσιαστικού». Στο Σχήμα 14 απεικονίζεται γραφικά ο σχηματισμός της δομής ουσιαστικού και ο τρόπος που συνδέονται τα στοιχεία της δομής μεταξύ τους.



Σχήμα 14: Αναπαράσταση δομής ουσιαστικού

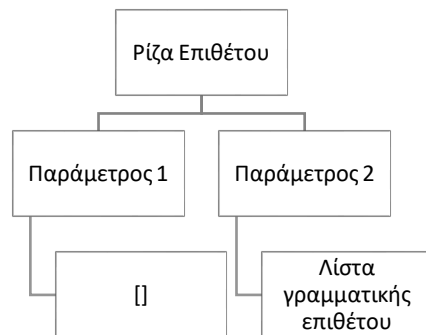
5.1.7.1.3 Δομή επιθετικού προσδιορισμού

Η Δομή Επιθετικού Προσδιορισμού αναπαριστά ένα επίθετο το οποίο προσδιορίζει την ιδιότητα ενός ουσιαστικού. Σε αντίθεση με τις προηγούμενες δομές, η δομή επιθετικού προσδιορισμού δεν συνδέεται σε άλλες (εκτός από αυτήν του ουσιαστικού) δομές και γι αυτό το λόγο η πρώτη παράμετρος είναι μια κενή λίστα. Η γενική μορφή αναπαράστασης της δομής επιθετικού προσδιορισμού είναι:

```

ΡίζαΕπιθέτου (
    [],
    Λίστα_Γραμματικής_Επιθέτου
)

```



Σχήμα 15: Αναπαράσταση δομής του επιθετικού προσδιορισμού

Στο Σχήμα 15 απεικονίζεται γραφικά ο σχηματισμός της δομής του επιθετικού προσδιορισμού.

5.1.7.1.4 Δομή Επιρρήματος

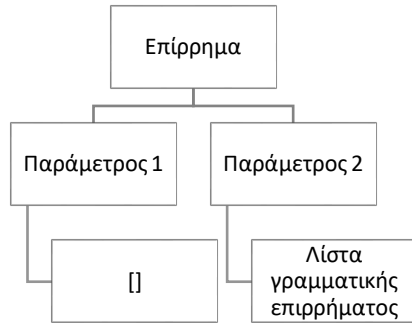
Η Δομή Επιρρήματος αναπαριστά ένα επίρρημα το οποίο προσδιορίζει ένα ρήμα. Σε αντίθεση με άλλες δομές, η δομή επιρρήματος δε συνδέεται σε άλλες (εκτός από αυτή του ρήματος) δομές και γι αυτό το λόγο η πρώτη παράμετρος είναι μια κενή λίστα. Η γενική μορφή αναπαράστασης της δομής επιρρήματος είναι:

```

Επίρρημα(
    [],
    Λίστα_Γραμματικής_Επιρρήματος
)

```

Στο Σχήμα 16 απεικονίζεται γραφικά ο σχηματισμός της δομής επιρρήματος.



Σχήμα 16: Αναπαράσταση δομής επιρρήματος

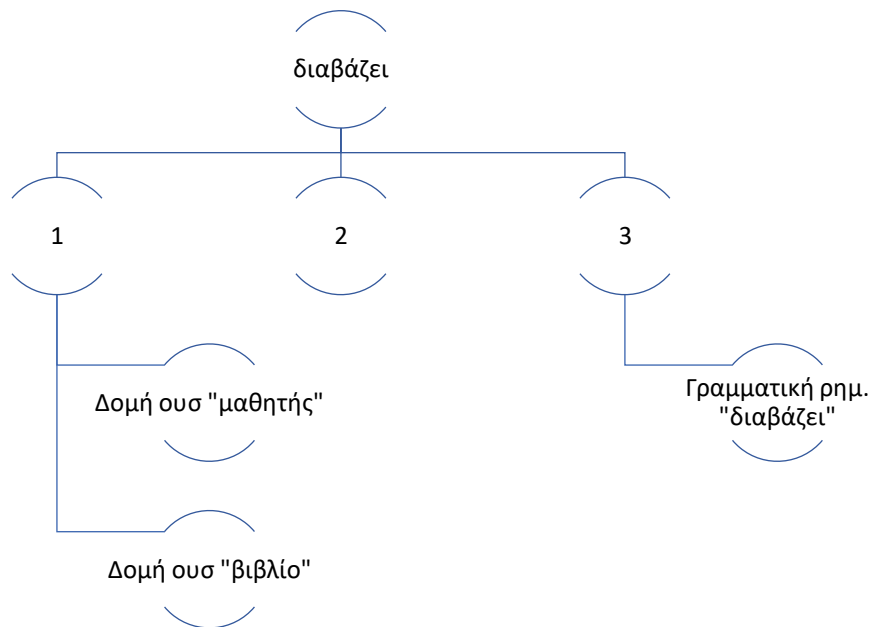
5.1.7.2 Παραδείγματα

5.1.7.2.1 Παράδειγμα 1

Πρόταση: «Ο μαθητής διαβάζει ένα βιβλίο»

Το ρήμα «διαβάζει» τοποθετείται στην κεφαλή της δομής της πρότασης η οποία περιέχει ως πρώτη παράμετρο μια λίστα με πρώτο στοιχείο το υποκείμενο και δεύτερο το αντικείμενο της πρότασης και είναι της μορφής [μαθητ(...), βιβλ(...)]. Η δεύτερη παράμετρος είναι μία λίστα με μοναδικό στοιχείο τη δομή του επιρρήματος, η οποία στην πρόταση είναι κενή εφόσον δεν υπάρχει επίρρημα. Η τρίτη παράμετρος είναι μια λίστα με τη γραμματική του ρήματος «διαβάζει»: [verb,διαβαζ,trans,enikos,trito,1].

Στο Σχήμα 17 φαίνεται η σχηματική αναπαράσταση της δομής πρότασης.



Σχήμα 17: Σχηματική αναπαράσταση της δομής της πρότασης «Ο μαθητής διαβάζει ένα βιβλίο»

Η αναπαράσταση της δομής της πρότασης «Ο μαθητής διαβάζει ένα βιβλίο» που παράγεται από την επεξεργασία της ως όρος Prolog φαίνεται παρακάτω:

διαβαζ (

```
[μαθητ ([], [noun, μαθητ, arseniko, enikos, onomastiki, 10]),
βιβλ ([], [noun, βιβλ, oydetero, enikos, aitiatiki, 2])],
[],
[verb, διαβαζ, trans, enikos, trito, 1]
)
```

5.1.7.2.2 Παράδειγμα 2

Πρόταση: «*Το λεμόνι δίνει νόστιμη γεύση στο φαγητό*»

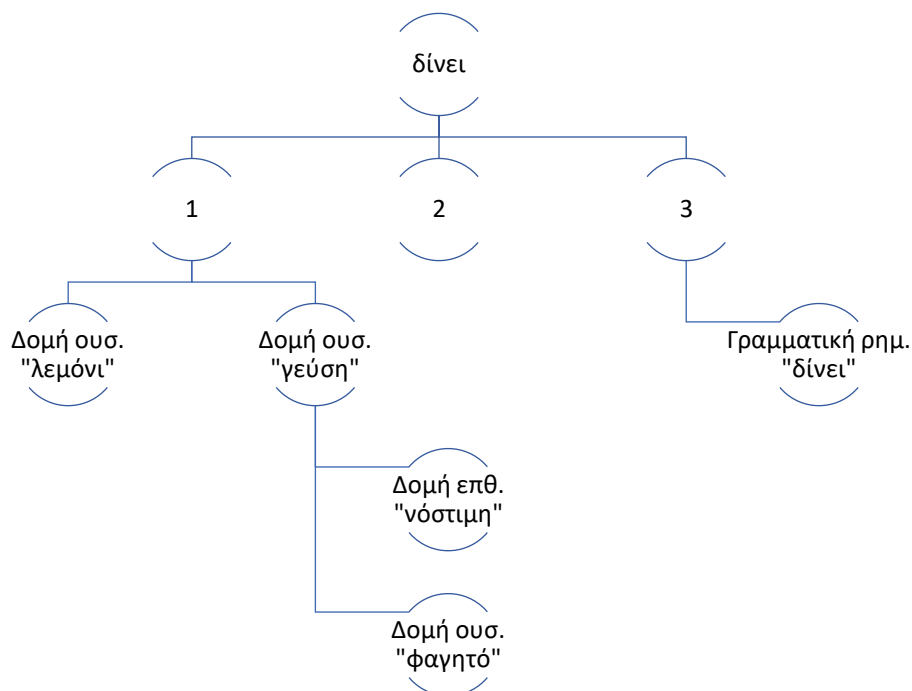
Το ρήμα «δίνει» τοποθετείται στην κεφαλή της δομής της πρότασης η οποία περιέχει ως πρώτη παράμετρο μια λίστα με πρώτο στοιχείο το υποκείμενο και δεύτερο το αντικείμενο της πρότασης και είναι της μορφής [λεμον(...), [γευσ(νοστιμ(...)...), φαγητ(...)]]. Το αντικείμενο είναι μία λίστα από τις δομές των ουσιαστικών «γεύση» και «φαγητό». Η δομή του ουσιαστικού «γεύση» περιέχει τη δομή του επιθετικού προσδιορισμού «νόστιμη» η οποία χαρακτηρίζει το ουσιαστικό. Η δεύτερη παράμετρος είναι μια λίστα με μοναδικό στοιχείο τη δομή του επιρρήματος, η οποία στην πρόταση είναι κενή εφόσον δεν υπάρχει επίρρημα. Η τρίτη παράμετρος είναι μια λίστα με τη γραμματική του ρήματος «δίνει»: [verb, διν, trans, enikos, trito, 1].

Στο Σχήμα 18 φαίνεται η σχηματική αναπαράσταση της δομής πρότασης.

Η αναπαράσταση της δομής της πρότασης «Το λεμόνι δίνει νόστιμη γεύση στο φαγητό» που παράγεται από την επεξεργασία της ως όρος Prolog φαίνεται παρακάτω:

διν (

```
[λεμον ([], [noun, λεμον, oydetero, enikos, onomastiki, 1]),
[γευσ([νοστιμ ([], [adjective, νοστιμ, thiliko, enikos, onomastiki, 1]),
φαγητ ([], [noun, φαγητ, oydetero, enikos, aitiatiki, 200])],
[noun, γευσ, thiliko, enikos, onomastiki, 101])]],
[],
[verb, διν, trans, enikos, trito, 1]
)
```



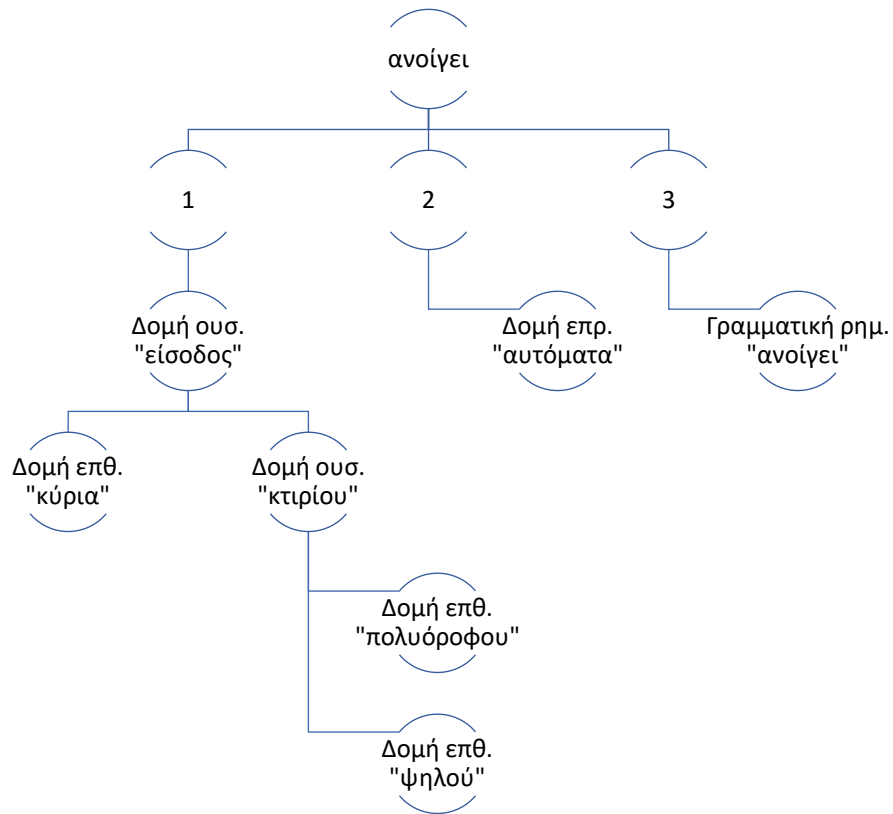
Σχήμα 18: Σχηματική αναπαράσταση της δομής της πρότασης «Το λεμόνι δίνει νόστιμη γεύση στο φαγητό»

5.1.7.2.3 Παράδειγμα 3

Πρόταση: «Η κύρια είσοδος του ψηλού πολυόροφου κτιρίου ανοίγει αυτόματα»

Το ρήμα «ανοίγει» τοποθετείται στην κεφαλή της δομής της πρότασης η οποία περιέχει ως πρώτη παράμετρο μια λίστα με μόνο στοιχείο το υποκείμενο της πρότασης καθώς η πρόταση δεν έχει αντικείμενο και είναι της μορφής [είσοδ([κτιρ([ψηλ(...), πολυοροφ(...)], ...), κυρι(...)], ...]. Η δομή του ουσιαστικού «είσοδος» περιέχει τη δομή του ουσιαστικού «κτιρίου» το οποίο περιέχει με τη σειρά του τη λίστα με τις δομές των επιθετικών προσδιορισμών «ψηλού» και «πολυόροφου» που χαρακτηρίζουν το ουσιαστικό «κτιρίου». Η δεύτερη παράμετρος είναι μία λίστα με μοναδικό στοιχείο τη δομή του επιρρήματος «αυτόματα» το οποίο προσδιορίζει το ρήμα «ανοίγει». Η τρίτη παράμετρος είναι μια λίστα με τη γραμματική του ρήματος «ανοίγει»: [verb, ανοιγ, intrans, enikos, trito, 1].

Στο Σχήμα 19 φαίνεται η σχηματική αναπαράσταση της δομής πρότασης.



Σχήμα 19: Σχηματική αναπαράσταση της δομής της πρότασης «Η κύρια είσοδος του ψηλού πολυόροφου κτιρίου ανοίγει αυτόματα».

Η αναπαράσταση της δομής της πρότασης «Η κύρια είσοδος του ψηλού πολυόροφου κτιρίου ανοίγει αυτόματα» που παράγεται από την επεξεργασία της ως όρος Prolog φαίνεται παρακάτω:

ανοιγ(

```
[εισοδ([κτιρ([ψηλ([], [adjective, ψηλ, ογdetero, enikos, geniki, 1]),
πολυοροφ([], [adjective, πολυοροφ, ογdetero, enikos, geniki, 1]),
[noun, κτιρ, ογdetero, enikos, geniki, 2]),
κυρι([], [adjective, κυρι, thiliko, enikos, onomastiki, 4])],
[noun, εισοδ, thiliko, enikos, onomastiki, 102])],
```

```
[αυτοματα([], [adverb, αυτοματα, tropiko])],
```

```
[verb, ανοιγ, intrans, enikos, trito, 1, attribute]
```

)

5.1.7.2.4 Παράδειγμα 4

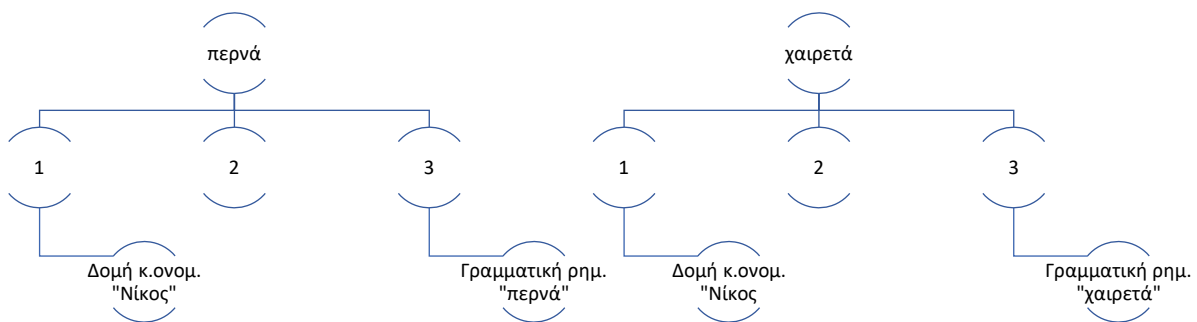
Πρόταση: «Ο Νίκος περνά και χαιρετά»

Η πρόταση έχει δύο ρήματα, έχει υποκείμενο αλλά όχι αντικείμενο. Επειδή το ρήμα είναι η κεφαλή της δομής πρότασης, θα δημιουργηθούν δύο δομές για να την αναπαραστήσουν, κάθε μία από τις οποίες θα έχει για κεφαλή ένα από τα δύο ρήματα της πρότασης.

Το ρήμα «περνά» τοποθετείται στην κεφαλή της δομής της πρώτης πρότασης η οποία περιέχει ως πρώτη παράμετρο μια λίστα με μόνο στοιχείο το υποκείμενο της πρότασης καθώς η πρόταση δεν έχει αντικείμενο και είναι της μορφής [Νικ(...)]. Η δεύτερη παράμετρος είναι μία λίστα με μοναδικό στοιχείο μια κενή λίστα εφόσον η πρόταση δεν έχει επίρρημα. Η τρίτη παράμετρος είναι μια λίστα με τη γραμματική του ρήματος «περνά»: [verb,περν,intrans,enikos,trito,2].

Για την αναπαράσταση της δομής της πρότασης με το δεύτερο ρήμα, δημιουργείται μια ξεχωριστή δομή όμοια με την πρώτη με τη διαφορά ότι η κεφαλή της είναι το ρήμα «χαιρετά» και στις παραμέτρους περιέχεται η γραμματική του. Η δομή της συνολικής πρότασης επιστρέφεται ως λίστα που περιέχει τις δύο δομές και είναι της μορφής [περν(Νικ(...)...),χαιρετ(Νικ(...)...)].

Στο Σχήμα 20 φαίνεται η σχηματική αναπαράσταση της δομής πρότασης:



Σχήμα 20: Σχηματική αναπαράσταση της δομής της πρότασης «Ο Νίκος περνά και χαιρετά»

Η αναπαράσταση της δομής της πρότασης «Ο Νίκος περνά και χαιρετά» που παράγεται από την επεξεργασία της ως όρος Prolog φαίνεται παρακάτω:

```
[  
    περν ([Νικ ([], [propernoun, Νικ, arseniko, enikos, onomastiki, 4])), [],  
    [verb, περν, intrans, enikos, trito, 2]),  
    χαιρετ ([Νικ ([], [propernoun, Νικ, arseniko, enikos, onomastiki, 4])), [],  
    [verb, χαιρετ, intrans, enikos, trito, 4])  
]
```

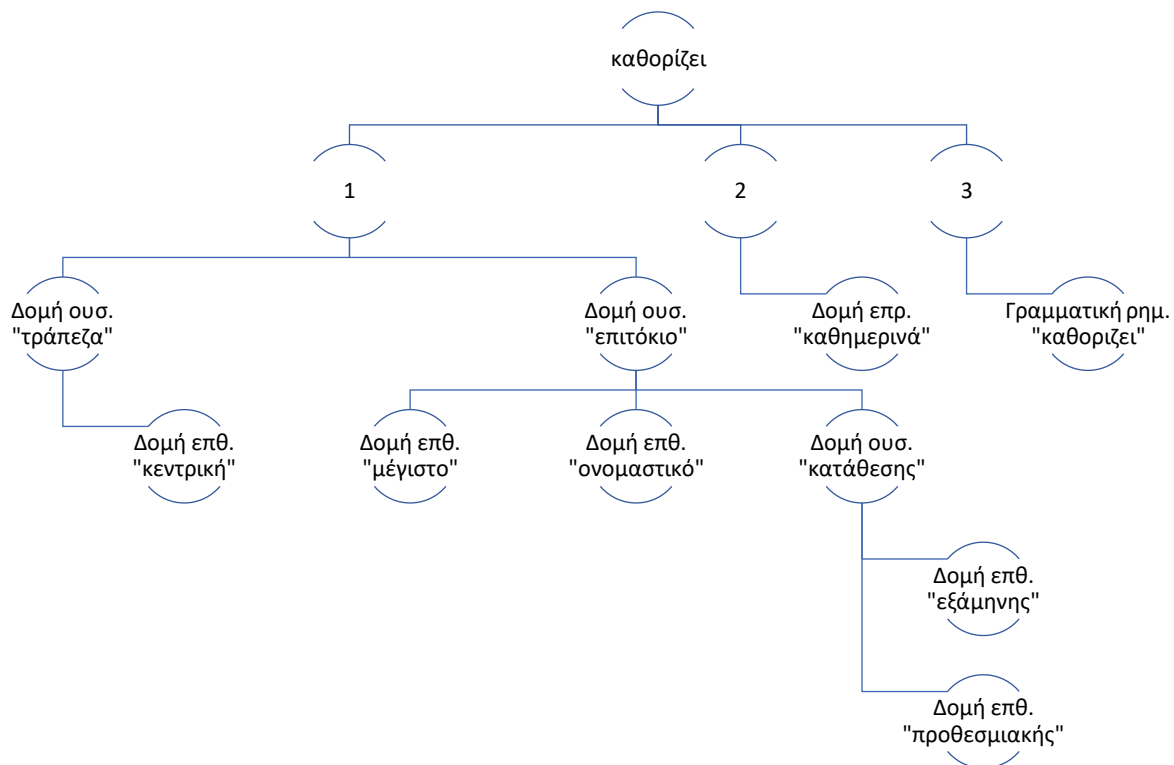
5.1.7.2.5 Παράδειγμα 5

Πρόταση: «*Η κεντρική τράπεζα καθορίζει το μέγιστο ονομαστικό επιτόκιο της εξάμηνης προθεσμιακής κατάθεσης καθημερινά*»

Η βασική πρόταση για τη δομή που σχηματίζεται είναι «*Η τράπεζα καθορίζει το επιτόκιο*». Το ουσιαστικό «*τράπεζα*» είναι το υποκείμενο και το ουσιαστικό «*επιτόκιο*» είναι το αντικείμενο. Το ουσιαστικό «*τράπεζα*» χαρακτηρίζεται από το επίθετο «*κεντρική*» και το ουσιαστικό «*επιτόκιο*» χαρακτηρίζεται από τα επίθετα «*μέγιστο*» και «*ονομαστικό*». Επίσης το ουσιαστικό «*επιτόκιο*» συνδέεται με το ουσιαστικό «*κατάθεσης*» το οποίο χαρακτηρίζεται από τα επίθετα «*εξάμηνης*» και «*προθεσμιακής*». Στην πρόταση προστίθεται το επίρρημα «*καθημερινά*» το οποίο χαρακτηρίζει το ρήμα «*καθορίζει*». Η δομή της πρότασης σε απλοποιημένη μορφή είναι η εξής:

καθοριζ(τραπεζ(κεντρικ()), επιτοκ(μεγιστ(), ονομαστικ()), καταθεσ(εξαμην, προθεσμιακ())), καθημεριν(), γραμματική)

Στο Σχήμα 21 φαίνεται η σχηματική αναπαράσταση της δομής πρότασης.



Σχήμα 21: Σχηματική αναπαράσταση της δομής της πρότασης «*Η κεντρική τράπεζα καθορίζει το μέγιστο ονομαστικό επιτόκιο της εξάμηνης προθεσμιακής κατάθεσης καθημερινά*»

Η αναπαράσταση της δομής της πρότασης «*Η κεντρική τράπεζα καθορίζει το μέγιστο ονομαστικό επιτόκιο της εξάμηνης προθεσμιακής κατάθεσης καθημερινά*» που παράγεται από την επεξεργασία της ως όρος Prolog φαίνεται παρακάτω:

καθοριζ(

```
[τραπεζ([κεντρικ([], [adjective, κεντρικ, thiliko, enikos, onomastiki, 1])), [
noun, τραπεζ, thiliko, enikos, onomastiki, 103]),
επιτοκ([μεγιστ([], [adjective, μεγιστ, oydetero, enikos, aitiatiki, 1]),
ονομαστ([], [adjective, ονομαστ, oydetero, enikos, aitiatiki, 1]),
καταθεσ([εξαμην([], [adjective, εξαμην, thiliko, enikos, geniki, 1]), προθεσμιακ([],
[adjective, προθεσμιακ, thiliko, enikos, geniki, 1]), [noun, καταθεσ, thiliko, enikos
, geniki, 100]), [noun, επιτοκ, oydetero, enikos, aitiatiki, 2])],
[καθημερινα([], [adverb, καθημερινα, χρονικο])],
[verb, καθοριζ, trans, enikos, trito, 1]
)
```

5.1.7.2.6 Παράδειγμα 6

Πρόταση: «*Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά*»

Επειδή η πρόταση έχει δύο ρήματα, το «είναι» και το «αποδεικνύει», δημιουργείται μία δομή για κάθε ρήμα και αναπαρίστανται οι εξής προτάσεις: «*Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα*» και «*Το μαθηματικό σύστημα αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά*».

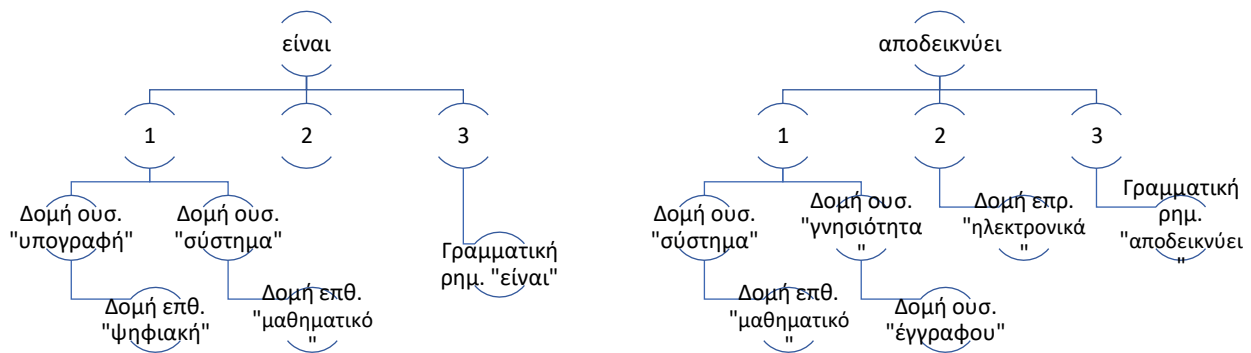
Η βασική πρόταση για τη δομή που σχηματίζεται από την πρώτη πρόταση είναι «*Η υπογραφή είναι σύστημα*». Το ουσιαστικό «υπογραφή» χαρακτηρίζεται από το επίθετο «ψηφιακή» και είναι το υποκείμενο της πρότασης ενώ το ουσιαστικό «σύστημα» χαρακτηρίζεται από το επίθετο «μαθηματικό» και είναι το αντικείμενο της πρότασης.

Η βασική πρόταση για τη δομή που σχηματίζεται από τη δεύτερη πρόταση είναι «*Το σύστημα αποδεικνύει τη γνησιότητα ηλεκτρονικά*». Το ουσιαστικό «σύστημα» χαρακτηρίζεται από το επίθετο «μαθηματικό» και είναι το υποκείμενο της πρότασης ενώ το ουσιαστικό «γνησιότητα» συνδέεται με το από το ουσιαστικό «εγγράφου» και είναι το αντικείμενο της πρότασης. Επιπλέον το επίρρημα «ηλεκτρονικά» προσδιορίζει το ρήμα «αποδεικνύει».

Η δομή της συνολικής πρότασης είναι μία λίστα που περιέχει δύο δομές, μία για τη δομή της πρώτης πρότασης και μία για τη δομή της δεύτερης πρότασης. Η απλοποιημένη μορφή της συνολικής δομής είναι:

```
[
ειν([υπογραφ(ψηφιακ()), συστημ(μαθηματικ())]),
αποδεικνυ(συστημ(μαθηματικ()), γνησιοτητ(εγγραφ()), [ηλεκτρονικα()])
]
```

Στο Σχήμα 22 φαίνεται η σχηματική αναπαράσταση της δομής πρότασης.



Σχήμα 22: Σχηματική αναπαράσταση της δομής της πρότασης «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά»

Η αναπαράσταση της δομής της πρότασης «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά» που παράγεται από την επεξεργασία της ως όρος Prolog φαίνεται παρακάτω:

```
[
  ειν (
    [υπογραφ ( [ψηφιακ ( [], [adjective, ψηφιακ, thiliko, enikos, onomastiki, 1
    ] ) ], [noun, υπογραφ, thiliko, enikos, onomastiki, 100] ) ,
    [συστημ ( [μαθηματικ ( [], [adjective, μαθηματικ, oydetero, enikos, aitiatiki, 1
    ] ) ], [noun, συστημ, oydetero, enikos, aitiatiki, 203] ) ] ] ,
    [ ] ,
    [verb, ειν, trans, enikos, trito, 10] ) ,
  αποδεικνυ (
    [συστημ ( [μαθηματικ ( [], [adjective, μαθηματικ, oydetero, enikos, aitiatiki, 1
    ] ) ], [noun, συστημ, oydetero, enikos, aitiatiki, 203] ) ,
    γνησιοτητ ( εγγραφ ( [], [noun, εγγραφ, oydetero, enikos, geniki, 200] ) , [noun, γνη
    σιοτητ, thiliko, enikos, aitiatiki, 6] ) ] ,
    [ηλεκτρονικα ( [], [adverb, ηλεκτρονικα, tropiko] ) ] ,
    [verb, αποδεικνυ, trans, enikos, trito, 1] )
]
```

5.1.7.2.7 Παράδειγμα 7

Πρόταση: «Οι μεγαλύτεροι πλανήτες του ηλιακού συστήματος είναι οι πλανήτες Δίας και Κρόνος».

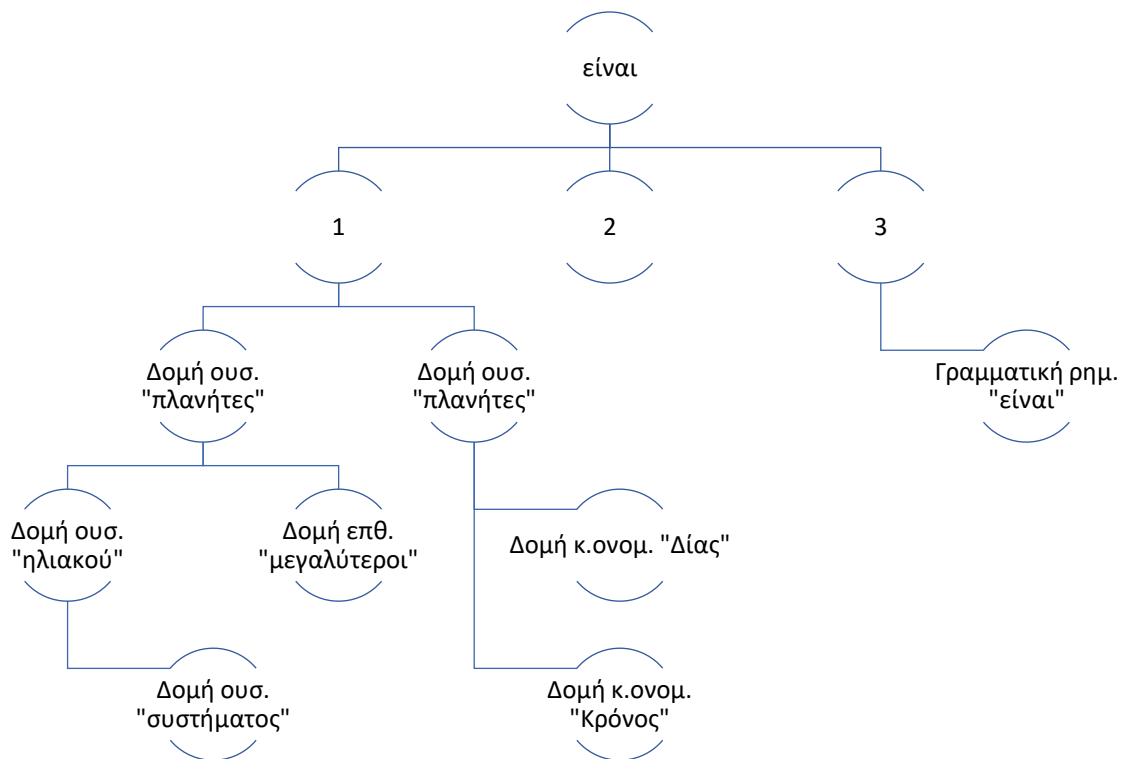
Η κεφαλή της δομής της πρότασης είναι το ρήμα «είναι». Το ουσιαστικό του υποκειμένου της πρότασης είναι το «πλανήτες» το οποίο χαρακτηρίζεται από το επίθετο «μεγαλύτεροι» και συνδέεται με το ουσιαστικό «συστήματος» το οποίο με τη σειρά του χαρακτηρίζεται από το επίθετο «ηλιακού». Το ουσιαστικό του αντικειμένου της πρότασης είναι επίσης το «πλανήτες» το οποίο συνδέεται με τα κύρια ονόματα «Δίας» και «Κρόνος».

Η απλοποιημένη μορφή της δομής είναι:

```

ειν (
  [πλανητ ([ηλιακ (συστημ ()), μεγαλυτερ ())], πλανητ ([Δι (), Κρον ()])],
  [],
  [verb, ειν, trans, enikos, trito, 10]
)
  
```

Στο Σχήμα 23 φαίνεται η σχηματική αναπαράσταση της δομής πρότασης.



Σχήμα 23: Σχηματική αναπαράσταση της δομής της πρότασης «Οι μεγαλύτεροι πλανήτες του ηλιακού συστήματος είναι οι πλανήτες Δίας και Κρόνος»

Η αναπαράσταση της δομής της πρότασης «Οι μεγαλύτεροι πλανήτες του ηλιακού συστήματος είναι οι πλανήτες Δίας και Κρόνος» που παράγεται από την επεξεργασία της ως όρος Prolog φαίνεται παρακάτω:

ειν (

```
[πλανητ ([ηλιακ ([συστημ ([], [noun, συστημ, οydetero, enikos, geniki, 203, entityType])), [noun, ηλιακ, arseniko, enikos, geniki, 11]), μεγαλυτερ ([], [adjective, μεγαλυτερ, arseniko, plithintikos, onomastiki, 1])], [noun, πλανητ, arseniko, plithintikos, onomastiki, 10]), πλανητ ([Δι ([], [propernoun, Δι, arseniko, enikos, onomastiki, 12, entity]), Κρον ([], [propernoun, Κρον, arseniko, enikos, onomastiki, 4, entity])], [noun, πλανητ, arseniko, enikos, onomastiki, 10])],
```

```
[],
```

```
[verb, ειν, trans, enikos, trito, 10]
```

)

5.1.7.2.8 Παράδειγμα 8

Πρόταση: «*Το μέγιστο ηλεκτρικό ρεύμα φόρτισης της μπαταρίας του φωτοβολταϊκού συστήματος που προτείνεται ανεπιφύλακτα αναφέρεται στο συνοδευτικό εγχειρίδιο*»

Η πρόταση έχει δύο ρήματα, το «προτείνεται» και το «αναφέρεται» οπότε δημιουργείται μία δομή για κάθε ρήμα και αναπαρίστανται οι εξής προτάσεις: «*Το μέγιστο ηλεκτρικό ρεύμα φόρτισης της μπαταρίας του φωτοβολταϊκού συστήματος προτείνεται ανεπιφύλακτα*» και «*Το μέγιστο ηλεκτρικό ρεύμα φόρτισης της μπαταρίας του φωτοβολταϊκού συστήματος αναφέρεται στο συνοδευτικό εγχειρίδιο*»

Η βασική πρόταση για τη δομή που σχηματίζεται από την πρώτη πρόταση είναι «*Το ρεύμα προτείνεται ανεπιφύλακτα*». Το ουσιαστικό «ρεύμα» του υποκειμένου χαρακτηρίζεται από τα επίθετα «μέγιστο» και «ηλεκτρικό» ενώ συνδέεται με το ουσιαστικό «φόρτισης». Το ουσιαστικό «φόρτισης» συνδέεται με το ουσιαστικό «μπαταρίας». Το ουσιαστικό «μπαταρίας» συνδέεται με το ουσιαστικό «συστήματος» το οποίο χαρακτηρίζεται από το επίθετο «φωτοβολταϊκού». Η πρόταση περιλαμβάνει το επίρρημα «ανεπιφύλακτα» το οποίο προσδιορίζει το ρήμα «προτείνεται».

Η βασική πρόταση για τη δομή που σχηματίζεται από τη δεύτερη πρόταση «*Το ρεύμα αναφέρεται στο εγχειρίδιο*». Η δομή του υποκειμένου είναι ακριβώς η ίδια με αυτήν της πρώτης πρότασης με ουσιαστικό το «ρεύμα». Το ουσιαστικό του αντικειμένου είναι το «εγχειρίδιο» το οποίο χαρακτηρίζεται από το επίθετο «συνοδευτικό».

Η δομή της συνολικής πρότασης είναι μία λίστα που περιέχει δύο δομές, μία για τη δομή της πρώτης πρότασης και μία για τη δομή της δεύτερης πρότασης. Η απλοποιημένη μορφή της συνολικής δομής είναι:

[

```

προτειν (
    ρευσμ ([μεγιστ ( ), ηλεκτρικ ( ), φορτισ (μπαταρ (φωτοβολταϊκ (συστημ ( ) ) ) ) ] )
    , ανεπιφύλακτα ( ) ) ,
αναφερ (
    ρευσμ ([μεγιστ ( ), ηλεκτρικ ( ), φορτισ (μπαταρ (φωτοβολταϊκ (συστημ ( ) ) ) ) ] )
    , εγχειριδ (συνοδευτικ ( ) )
]

```

Στο Σχήμα 24 φαίνεται η σχηματική αναπαράσταση της δομής πρότασης.

Η αναπαράσταση της δομής της πρότασης «*Το μέγιστο ηλεκτρικό ρεύμα φόρτισης της μπαταρίας του φωτοβολταϊκού συστήματος που προτείνεται ανεπιφύλακτα αναφέρεται στο συνοδευτικό εγχειρίδιο*» που παράγεται από την επεξεργασία της ως όρος Prolog φαίνεται παρακάτω:

```

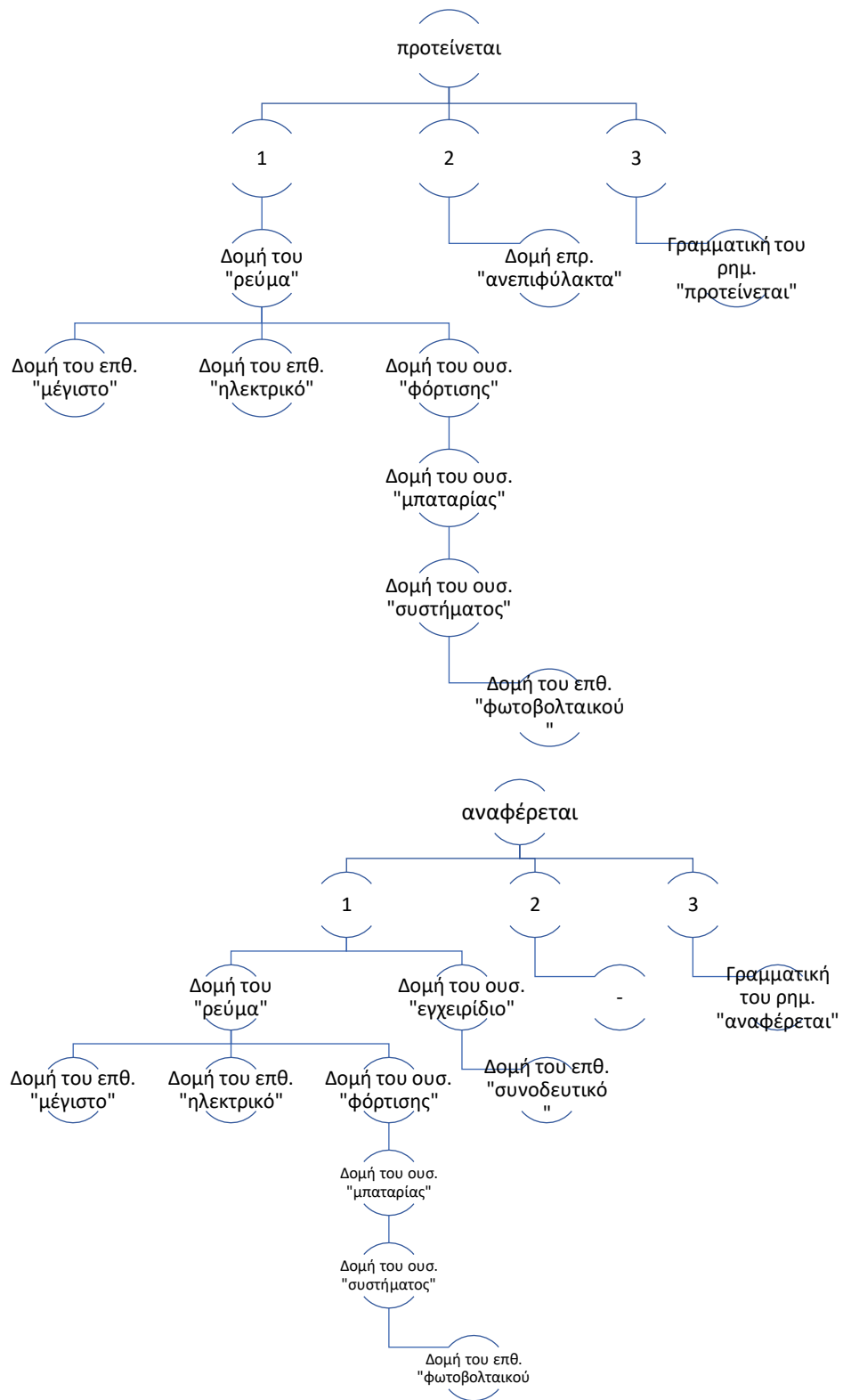
[
    προτειν ([ρευσμ ([μεγιστ ([ ], [adjective, μεγιστ, οydetero, enikos, aitiatiki, 1]
    ), ηλεκτρικ ([ ], [adjective, ηλεκτρικ, οydetero, enikos, aitiatiki, 1] ) , φορτισ ([μπαταρ
    ([φωτοβολταϊκ ([συστημ ([ ], [noun, συστημ, οydetero, enikos, geniki, 203, entityType]
    ) ] , [noun, φωτοβολταϊκ, οydetero, enikos, geniki, 200] ) ] , [noun, μπαταρ, thiliko, eniko
    s, geniki, 8] ) ] , [noun, φορτισ, thiliko, enikos, geniki, 101] ) ] , [noun, ρευσμ, οydetero, e
    nikos, aitiatiki, 203] ) ] , [ανεπιφύλακτα ([ ], [adverb, ανεπιφύλακτα, tropiko] ) ] , [verb
    , προτειν, intrans, enikos, trito, 12, attribute] ) ,

    αναφερ ([ρευσμ ([μεγιστ ([ ], [adjective, μεγιστ, οydetero, enikos, aitiatiki, 1] )
    , ηλεκτρικ ([ ], [adjective, ηλεκτρικ, οydetero, enikos, aitiatiki, 1] ) , φορτισ ([μπαταρ
    ([φωτοβολταϊκ ([συστημ ([ ], [noun, συστημ, οydetero, enikos, geniki, 203, entityType]
    ) ] , [noun, φωτοβολταϊκ, οydetero, enikos, geniki, 200] ) ] , [noun, μπαταρ, thiliko, enikos
    , geniki, 8] ) ] , [noun, φορτισ, thiliko, enikos, geniki, 101] ) ] , [noun, ρευσμ, οydetero, en
    ikos, aitiatiki, 203] ) ] , [εγχειριδ ([συνοδευτικ ([ ], [adjective, συνοδευτικ, οydetero,
    enikos, aitiatiki, 1] ) ] , [noun, εγχειριδ, οydetero, enikos, aitiatiki, 2] ) ] ) ] ,

    [ ],

    [verb, αναφερ, trans, enikos, trito, 12] )
]

```



Σχήμα 24: Σχηματική αναπαράσταση της δομής της πρότασης «Το μέγιστο ηλεκτρικό ρεύμα φόρτισης της μπαταρίας του φωτοβολταϊκού συστήματος που προτείνεται ανεπιφύλακτα αναφέρεται στο συνοδευτικό εγχειρίδιο»

Στα επόμενα παραδείγματα φαίνεται η απλουστευμένη μορφή της αναπαράστασης απλών και σύνθετων προτάσεων. Στην αναπαράσταση εμφανίζονται μόνο τα στοιχεία της δομής που αποτελούν μέρη του λόγου και έχουν αφαιρεθεί οι υπόλοιπες παράμετροι για απλότητα.

- «Ο μαθητής περνά»
περν (μαθητ ())
- «Ο μαθητής περνά γρήγορα»
περν ([μαθητ ()] , γρηγορα ())
- «Ο μαθητής Γιάννης περνά γρήγορα»
περν (Γιανν (μαθητ ()) , γρηγορα ())
- «Ο μαθητής διαβάζει ένα βιβλίο»
διαβαζ (μαθητ () , βιβλ ())
- «Ο καλός μαθητής διαβάζει ένα μεγάλο βιβλίο»
διαβαζ (μαθητ (καλ ()) , βιβλ (μεγαλ ()))
- «Ο καλός ψηλός μαθητής διαβάζει ένα μεγάλο διδακτικό βιβλίο»
διαβαζ (μαθητ (καλ ()) , βιβλ (μεγαλ ()))
- «Ο καλός ψηλός μαθητής διαβάζει ένα μεγάλο διδακτικό βιβλίο»
διαβαζ (μαθητ (καλ ()) , βιβλ (μεγαλ () , διδακτικ ()))
- «Ο καλός ψηλός μαθητής διαβάζει ένα μεγάλο διδακτικό βιβλίο σήμερα»
διαβαζ (μαθητ (καλ ()) , βιβλ (μεγαλ () , διδακτικ ()) , σήμερα ())

Η ακριβής αναπαράσταση των προηγούμενων παραδειγμάτων ως όρων της Prolog φαίνεται παρακάτω.

- «Ο μαθητής περνά»
περν ([μαθητ ([] , [noun, μαθητ, arseniko, enikos, onomastiki, 10])] ,
[] ,
[verb, περν, intrans, enikos, trito, 2])
- «Ο μαθητής περνά γρήγορα»
περν ([μαθητ ([] , [noun, μαθητ, arseniko, enikos, onomastiki, 10])] ,
[γρήγορα ([] , [adverb, γρηγορα, χρονικο])] ,
[verb, περν, intrans, enikos, trito, 2])
- «Ο μαθητής Γιάννης περνά γρήγορα»

περν ([Γιανν (μαθητ ([], [noun, μαθητ, arseniko, enikos, onomastiki, 10]), [pr
opernoun, Γιανν, arseniko, enikos, onomastiki, 3, entity0])], [
γρηγορα ([], [adverb, γρηγορα, χρονικο])], [
verb, περν, intrans, enikos, trito, 2])

- «Ο μαθητής διαβάζει ένα βιβλίο»

διαβαζ ([μαθητ ([], [noun, μαθητ, arseniko, enikos, onomastiki, 10]), βιβλ ([
, [noun, βιβλ, oydetero, enikos, aitiatiki, 2])], [
, [
verb, διαβαζ, trans, enikos, trito, 1])

- «Ο καλός μαθητής διαβάζει ένα μεγάλο βιβλίο»

διαβαζ ([μαθητ ([καλ ([], [adjective, καλ, arseniko, enikos, onomastiki, 1])]
, [noun, μαθητ, arseniko, enikos, onomastiki, 10]), βιβλ ([μεγαλ ([], [adjective,
μεγαλ, oydetero, enikos, aitiatiki, 1,])], [noun, βιβλ, oydetero, enikos, aitiat
iki, 2])], [
, [
verb, διαβαζ, trans, enikos, trito, 1])

- «Ο καλός ψηλός μαθητής διαβάζει ένα μεγάλο διδακτικό βιβλίο»

διαβαζ ([μαθητ ([καλ ([], [adjective, καλ, arseniko, enikos, onomastiki, 1])],
ψηλ ([], [adjective, ψηλ, arseniko, enikos, onomastiki, 1,])], [noun, μαθητ, arse
niko, enikos, onomastiki, 10]), βιβλ ([μεγαλ ([], [adjective, μεγαλ, oydetero, en
ikos, aitiatiki, 1,])], διδακτικ ([], [adjective, διδακτικ, oydetero, enikos, ait
iatiki, 1,])], [noun, βιβλ, oydetero, enikos, aitiatiki, 2])], [
, [
verb, διαβαζ, trans, enikos, trito, 1])

- «Ο καλός ψηλός μαθητής διαβάζει ένα μεγάλο διδακτικό βιβλίο σήμερα»

διαβαζ ([μαθητ ([καλ ([], [adjective, καλ, arseniko, enikos, onomastiki, 1])],
ψηλ ([], [adjective, ψηλ, arseniko, enikos, onomastiki, 1,])], [noun, μαθητ, arse
niko, enikos, onomastiki, 10]), βιβλ ([μεγαλ ([], [adjective, μεγαλ, oydetero, en
ikos, aitiatiki, 1,])], διδακτικ ([], [adjective, διδακτικ, oydetero, enikos, ait
iatiki, 1,])], [noun, βιβλ, oydetero, enikos, aitiatiki, 2])], [
, [
verb, διαβαζ, trans, enikos, trito, 1])

- «Ο καλός ψηλός μαθητής διαβάζει ένα μεγάλο διδακτικό βιβλίο σήμερα»

διαβαζ ([μαθητ ([καλ ([], [adjective, καλ, arseniko, enikos, onomastiki, 1])],
ψηλ ([], [adjective, ψηλ, arseniko, enikos, onomastiki, 1,])], [noun, μαθητ, arse
niko, enikos, onomastiki, 10]), βιβλ ([μεγαλ ([], [adjective, μεγαλ, oydetero, en
ikos, aitiatiki, 1,])], διδακτικ ([], [adjective, διδακτικ, oydetero, enikos, ait
iatiki, 1,])], [noun, βιβλ, oydetero, enikos, aitiatiki, 2])], [
σημερα ([], [adverb, σημερα, χρονικο])], [
verb, διαβαζ, trans, enikos, trito, 1])

5.2 Εξαγωγή οντοτήτων και συσχετίσεων

Η δημιουργία διαγραμμάτων οντοτήτων-συσχετίσεων (Entity-Relationships diagrams) είναι ένας τρόπος γραφικής αναπαράστασης οντοτήτων, σχέσεων και ιδιοτήτων (entities, relations, attributes). Τις τελευταίες δεκαετίες έχει διεξαχθεί έρευνα στον τρόπο αυτόματης αναπαράστασης προτάσεων της φυσικής γλώσσας σε διαγράμματα οντοτήτων-συσχετίσεων και έχουν προταθεί κανόνες μετατροπής. Η έρευνα έχει διεξαχθεί κυρίως με αναφορά στην αγγλική γλώσσα.

Οι πρώτοι κανόνες διατυπώθηκαν το 1976 από τον Peter Chen ο οποίος ήταν από τους πρωτοπόρους της χρήσης οντοτήτων και σχέσεων στη μοντελοποίηση και το σχεδιασμό πληροφοριακών συστημάτων. Οι κανόνες αυτοί χρησιμοποιούνται μέχρι σήμερα ως οδηγίες του μοντέλου οντοτήτων - συσχετίσεων (ER Model), ενώ εμπλουτίστηκαν με ιδέες που προτάθηκαν μεταγενέστερα [3] [37] [38] . Στην πραγματικότητα οι κανόνες είναι ένα σύνολο από κατευθυντήριες γραμμές για τη μετάβαση από τη γραμματική μιας πρότασης σε οντότητες και συσχετίσεις μεταξύ τους. Οι κανόνες δεν είναι δυνατόν να καλύψουν όλες τις δυνατές περιπτώσεις, γι αυτό σε αρκετές δημοσιεύσεις αναφέρεται ότι το μοντέλο δεν είναι εντελώς αυτόματο και ενδέχεται να απαιτηθεί η παρέμβαση του χρήστη κατά περίπτωση. Στην διατριβή αυτή αναγνωρίζεται και χρησιμοποιείται ένα υποσύνολο από τα βασικά σύμβολα του μοντέλου που αποτελούν οι *οντότητες (entities)*, οι *τύποι οντοτήτων (entity types)*, οι *ιδιότητες οντοτήτων (attributes of the entities)*, οι *σχέσεις των οντοτήτων (relationships of the entities)* και οι *ιδιότητες των οντοτήτων (attributes of the entities)*.

Η κεντρική ιδέα για τη δημιουργία ενός διαγράμματος οντοτήτων-συσχετίσεων από κείμενο γραμμένο σε φυσική γλώσσα είναι η αντιστοίχιση βασικών μερών του λόγου σε στοιχεία του μοντέλου, δηλαδή οντότητες, σχέσεις και ιδιότητες. Συγκεκριμένα τα **ουσιαστικά** αντιστοιχούν σε *τύπους οντοτήτων (entity types)*, τα **κύρια ονόματα** και γενικά τα **ονόματα** σε *οντότητες (entities)*, τα **επίθετα** σε *ιδιότητες οντοτήτων (attributes of entities)*, τα **μεταβατικά ρήματα** σε *σχέσεις των οντοτήτων (relationships of entities)* και τα **αμετάβατα ρήματα** σε *ιδιότητες των οντοτήτων (attributes of the entities)*. Σε αυτούς τους κανόνες προστίθενται επιπλέον κανόνες που σχετίζονται με τη γραμματική της πρότασης. Έτσι π.χ. στη φράση «ο μαθητής Γιάννης», το «μαθητής» χρησιμοποιείται ως ιδιότητα οντότητας αντί ως οντότητα παρά το γεγονός ότι είναι ουσιαστικό.

Στην διατριβή μας χρησιμοποιήθηκαν πολλοί από τους κανόνες που έχουν προταθεί στη βιβλιογραφία για την αυτόματη δημιουργία από προτάσεις του διαγράμματος οντοτήτων – συσχετίσεων. Σε αυτούς έχουν προστεθεί άλλοι που προτείνονται στα πλαίσια αυτής της διατριβής και αφορούν στην ελληνική γλώσσα. Η διατύπωση όλων των κανόνων μετατροπής που χρησιμοποιούνται γίνεται σε επόμενη παράγραφο.

5.2.1 Πίνακες συμβόλων

Για την αναπαράσταση των στοιχείων του μοντέλου οντοτήτων - συσχετίσεων έχουν προταθεί μετά την εισαγωγή του το 1976 διάφορες διαγραμματικές τεχνικές. Στη συγκεκριμένη διατριβή κάθε στοιχείο αναπαριστάται με συγκεκριμένο σχήμα που χρησιμοποιείται στη βιβλιογραφία και επιπλέον για κάθε σχήμα έχει αντιστοιχηθεί ένα διαφορετικό χρώμα ώστε να διακρίνονται ευκολότερα τα στοιχεία στα διαγράμματα.

Στον Πίνακα 9 φαίνεται η αναπαράσταση κάθε στοιχείου του μοντέλου οντοτήτων-συσχετίσεων με το αντίστοιχο σχήμα που εμφανίζεται στην διατριβή.

Στοιχείο μοντέλου οντοτήτων-συσχετίσεων (ER Model Component)	Χρώμα-Σχήμα	Σύμβολο
Τύπος οντότητας (Entity type)	Μπλε τετράγωνο	
Ιδιότητα οντότητας (Attribute of an entity)	Πράσινο οβάλ	
Σχέση οντοτήτων (Relationship)	Μωβ ρόμβος	
Ιδιότητα σχέσης (Attribute of relationship)	Πορτοκαλί οκτάγωνο	
Οντότητα (Entity)	Γκρι τετράγωνο	

Πίνακας 9. Τα σχήματα αναπαράστασης κάθε στοιχείου του μοντέλου οντοτήτων-συσχετίσεων

Από	Σε	Δηλώνει	Αναπαράσταση
Τύπος Οντότητας/ Οντότητα (Entity type/Entity)	Σχέση (Relationship)	Ο Τύπος Οντότητας/ Οντότητα έχει θέση Υποκειμένου	
Σχέση (Relationship)	Τύπος Οντότητας / Οντότητα (Entity type/Entity)	Ο Τύπος Οντότητας/ Οντότητα έχει θέση Αντικειμένου	
Ιδιότητα Τύπου Οντότητας/ Οντότητας (Attribute of an Entity Type/Entity)	Τύπος Οντότητας / Οντότητα (Entity Type/Entity)	Ο Τύπος Οντότητας/ Οντότητα έχει Ιδιότητα	
Ιδιότητα σχέσης (Attribute of relationship)	Σχέση (Relationship)	Η Σχέση έχει Ιδιότητα σχέσης	
Τύπος Οντότητας/ Οντότητα (Entity type/Entity)	Ιδιότητα οντότητας (Attribute of an entity)	Ο Τύπος Οντότητας/ Οντότητα έχει θέση Υποκειμένου και έχει Ιδιότητα Οντότητας	
Ιδιότητα (Attribute)	Ιδιότητα σχέσης (Attribute of relationship)	Η Ιδιότητα έχει Ιδιότητα σχέσης	

Πίνακας 10. Η αναπαράσταση των συνδέσεων μεταξύ των στοιχείων του μοντέλου οντοτήτων-συσχετίσεων και η σημασία τους

Οι συνδέσεις μεταξύ των στοιχείων παριστάνονται με γραμμές συγκεκριμένης κατεύθυνσης η οποία δηλώνει τον τρόπο της μεταξύ τους συσχέτισης. Μία σύνδεση που ξεκινάει από μία *οντότητα ή τύπο οντότητας* και καταλήγει σε μία *σχέση* δηλώνει ότι **η οντότητα ή ο τύπος οντότητας είναι υποκείμενο στην πρόταση**. Αντίστοιχα μία σύνδεση που ξεκινάει από μία *σχέση* και καταλήγει σε μία *οντότητα ή τύπο οντότητας* δηλώνει ότι **η οντότητα ή ο τύπος οντότητας είναι αντικείμενο στην πρόταση**. Μία σύνδεση που ξεκινάει από μία *ιδιότητα* και καταλήγει σε μία *οντότητα ή τύπο οντότητας* δηλώνει ότι **η ιδιότητα αναφέρεται στη συγκεκριμένη οντότητα ή τύπο οντότητας**. Αντίστοιχα μία σύνδεση που ξεκινάει από μία *ιδιότητα σχέσης* και καταλήγει σε μία *σχέση* δηλώνει ότι **η ιδιότητα αναφέρεται στη συγκεκριμένη σχέση**. Μία σύνδεση που ξεκινάει από μία *ιδιότητα σχέσης* και καταλήγει σε μία άλλη *ιδιότητα* δηλώνει ότι **η ιδιότητα σχέσης αναφέρεται στη συγκεκριμένη ιδιότητα**.

Στον Πίνακα 10 φαίνεται η αναπαράσταση των συνδέσεων μεταξύ των στοιχείων του μοντέλου οντοτήτων-συσχετίσεων και η σημασία τους.

5.2.2 Κανόνες του μοντέλου οντοτήτων-συσχετίσεων

Το Μοντέλο Οντοτήτων – Συσχετίσεων δημιουργεί τα στοιχεία του διαγράμματος Οντοτήτων – Συσχετίσεων με βάση το μέρος του λόγου στο οποίο ανήκουν οι λέξεις της πρότασης. Για τη μετατροπή χρησιμοποιείται το σύνολο των κανόνων που ακολουθούν.

ΚΑΝΟΝΑΣ 1. Κάθε **ουσιαστικό** αντιστοιχεί σε **τύπο οντότητας** (*entity type*) στο ER διάγραμμα εκτός αν ισχύει κάποιος κανόνας (από τους επόμενους) σύμφωνα με τον οποίο αντιστοιχεί σε *ιδιότητα οντότητας*.

ΚΑΝΟΝΑΣ 2. Κάθε **μεταβατικό ρήμα** (*transitive verb*) αντιστοιχεί σε **σχέση** (*relationship*) ανάμεσα σε *τύπους οντοτήτων ή οντότητες* στο ER διάγραμμα.

Παράδειγμα

Πρόταση: «Ο μαθητής αναπτύσσει δεξιότητες»

Οι λέξεις «μαθητής» και «δεξιότητες» είναι ουσιαστικά και αντιστοιχούν σε *τύπους οντοτήτων*. Η λέξη «αναπτύσσει» είναι μεταβατικό ρήμα και αντιστοιχεί σε *σχέση*. Ο τύπος οντότητας «μαθητής» συνδέεται στη σχέση «αναπτύσσει» ως υποκείμενο ενώ ο τύπος οντότητας «δεξιότητες» συνδέεται στη σχέση ως αντικείμενο. Το ER διάγραμμα που δημιουργείται από την πρόταση φαίνεται παρακάτω.



ΚΑΝΟΝΑΣ 3. Τα **κύρια ονόματα** (*proper nouns*) και γενικά τα **ονόματα** αντιστοιχούν σε *οντότητες* (*entities*) στο ER διάγραμμα.

Παράδειγμα

Πρόταση: «Η Μαρία διδάσκει φυσική»

Το κύριο όνομα «Μαρία» αντιστοιχεί σε *οντότητα*, το μεταβατικό ρήμα «διδάσκει» αντιστοιχεί σε *σχέση* και το ουσιαστικό «φυσική» αντιστοιχεί σε *τύπο οντότητας*. Η οντότητα «Μαρία» συνδέεται στη σχέση «διδάσκει» ως «υποκείμενο» ενώ ο τύπος οντότητας «φυσική» συνδέεται στη σχέση ως αντικείμενο. Το ER διάγραμμα που δημιουργείται από στην πρόταση φαίνεται παρακάτω.



ΚΑΝΟΝΑΣ 4. Κάθε *επίθετο* (adjective) αντιστοιχεί σε *ιδιότητα οντότητας* (attribute of an entity) στο ER διάγραμμα.

Παράδειγμα

Πρόταση: «Ο επιμελής μαθητής αναπτύσσει αξιόλογες δεξιότητες»

Οι λέξεις «μαθητής» και «δεξιότητες» είναι ουσιαστικά και αντιστοιχούν σε *τύπους οντότητας*. Η λέξη «αναπτύσσει» είναι μεταβατικό ρήμα και αντιστοιχεί σε *σχέση*. Οι λέξεις «επιμελής» και «αξιόλογες» είναι επίθετα και αντιστοιχούν σε *ιδιότητες οντοτήτων*. Ο τύπος οντότητας «μαθητής» συνδέεται στη σχέση «αναπτύσσει» ως υποκείμενο ενώ ο τύπος οντότητας «δεξιότητες» συνδέεται στη σχέση ως αντικείμενο. Οι ιδιότητες οντοτήτων «επιμελής» και «αξιόλογες» συνδέονται στους τύπους οντοτήτων «μαθητής» και «δεξιότητες» αντίστοιχα. Το ER διάγραμμα που δημιουργείται από την πρόταση φαίνεται παρακάτω.

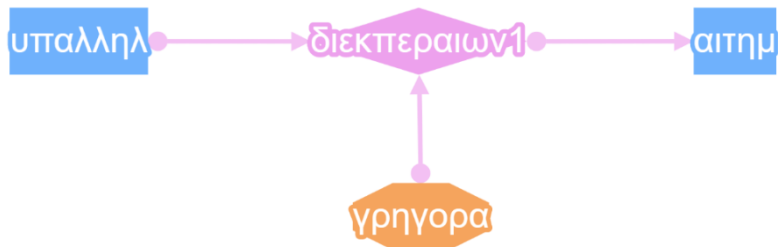


ΚΑΝΟΝΑΣ 5. Κάθε *επίρρημα* (adverb) αντιστοιχεί σε *ιδιότητα σχέσης* (attribute of relationship) στο ER διάγραμμα.

Παράδειγμα

Πρόταση: «Ο υπάλληλος διεκπεραιώνει το αίτημα γρήγορα»

Οι λέξεις «υπάλληλος» και «αίτημα» είναι ουσιαστικά και αντιστοιχούν σε *τύπους οντοτήτων*. Η λέξη «διεκπεραιώνει» είναι μεταβατικό ρήμα και αντιστοιχεί σε *σχέση*. Η λέξη «γρήγορα» είναι επίρρημα και αντιστοιχεί σε *ιδιότητα σχέσης*. Ο τύπος οντότητας «υπάλληλος» συνδέεται στη σχέση «διεκπεραιώνει» ως υποκείμενο ενώ ο τύπος οντότητας «αίτημα» συνδέεται στη σχέση ως αντικείμενο. Η ιδιότητα σχέσης «γρήγορα» συνδέεται στη σχέση «διεκπεραιώνει». Το ER διάγραμμα που δημιουργείται από την πρόταση φαίνεται παρακάτω.



ΚΑΝΟΝΑΣ 6. Κάθε **αμετάβατο ρήμα** (*intransitive verb*) αντιστοιχεί σε **ιδιότητα οντότητας** (*attribute of an entity*) στο ER διάγραμμα.

Παράδειγμα 1

Πρόταση: «Ο χρόνος περνά»

Οι λέξη «χρόνος» είναι ουσιαστικό και αντιστοιχεί σε *τύπο οντότητας*. Η λέξη «περνά» είναι αμετάβατο ρήμα και αντιστοιχεί σε *ιδιότητα οντότητας*. Η ιδιότητα οντότητας «περνά» συνδέεται με τον τύπο οντότητας «χρόνος». Επειδή η ιδιότητα οντότητας «περνά» είναι μια ενέργεια του τύπου οντότητας «χρόνος», ο τύπος οντότητας «χρόνος» είναι συνδεδεμένος ως υποκείμενο στην ιδιότητα οντότητας «περνά». Το ER διάγραμμα που δημιουργείται από στην πρόταση φαίνεται παρακάτω



Παράδειγμα 2

Πρόταση: «Ο χρόνος περνά γρήγορα»

Οι λέξη «χρόνος» είναι ουσιαστικό και αντιστοιχεί σε *τύπο οντότητας*. Η λέξη «περνά» είναι αμετάβατο ρήμα και αντιστοιχεί σε *ιδιότητα οντότητας*. Η λέξη «γρήγορα» είναι επίρρημα και αντιστοιχεί σε

ιδιότητα σχέσης. Η ιδιότητα οντότητας «περνά» συνδέεται στον τύπο οντότητας «χρόνος». Επειδή η ιδιότητα οντότητας «περνά» είναι μια ενέργεια του τύπου οντότητας «χρόνος», ο τύπος οντότητας «χρόνος» είναι συνδεδεμένος ως υποκείμενο στην ιδιότητα οντότητας «περνά». Η ιδιότητα σχέσης «γρήγορα» συνδέεται στην ιδιότητα οντότητας «περνά». Το ER διάγραμμα που δημιουργείται από στην πρόταση φαίνεται παρακάτω.



ΚΑΝΟΝΑΣ 7. Αν η πρόταση έχει τη μορφή «**Το Χ είναι Υ**» με το Χ να είναι ουσιαστικό και το Υ να είναι επίθετο, τότε το Χ αντιστοιχεί σε **τύπο οντότητας** και το Υ αντιστοιχεί σε **ιδιότητα οντότητας**.

Παράδειγμα

Πρόταση: «*Το έγγραφο είναι εμπιστευτικό*»

Οι λέξη «έγγραφο» είναι ουσιαστικό και αντιστοιχεί σε *τύπο οντότητας*. Η λέξη «εμπιστευτικό» είναι επίθετο και αντιστοιχεί σε *ιδιότητα οντότητας*. Η ιδιότητα οντότητας «εμπιστευτικό» συνδέεται στον τύπο οντότητας «έγγραφο». Το ER διάγραμμα που δημιουργείται από στην πρόταση φαίνεται παρακάτω.



ΚΑΝΟΝΑΣ 8. Αν ένα *ουσιαστικό* προηγείται ενός *κύριου ονόματος*, το ουσιαστικό αντιστοιχεί σε *ιδιότητα οντότητας*.

Παράδειγμα

Πρόταση: «*Η μαθήτρια Άννα διαβάζει ιστορία*»

Η λέξη «Άννα» είναι κύριο όνομα και αντιστοιχεί σε *οντότητα*. Οι λέξη «μαθήτρια» είναι ουσιαστικό και αντιστοιχεί σε τύπο οντότητας. Ο τύπος οντότητας «μαθήτρια» συνδέεται στην οντότητα «Μαρία» ως ιδιότητα οντότητας. Το ER διάγραμμα που δημιουργείται από στην πρόταση φαίνεται παρακάτω.



ΚΑΝΟΝΑΣ 9. Αν ένα *ουσιαστικό* προηγείται ενός *άλλου ουσιαστικού*, το πρώτο αντιστοιχεί σε *τύπο οντότητας* ενώ το δεύτερο ουσιαστικό αντιστοιχεί σε *ιδιότητα οντότητας*.

Παράδειγμα

Πρόταση: «*Η λέξη κλειδί διευκολύνει την αναζήτηση*»

Οι λέξεις «λέξη» και «κλειδί» είναι ουσιαστικά, από τα οποία το πρώτο («λέξη») αντιστοιχεί σε *τύπο οντότητας* ενώ το δεύτερο ουσιαστικό («κλειδί») αντιστοιχεί σε *ιδιότητα οντότητας* γιατί χαρακτηρίζει το πρώτο. Η λέξη «αναζήτηση» είναι επίσης ουσιαστικό και αντιστοιχεί σε *τύπο οντότητας*. Η λέξη «διευκολύνει» είναι μεταβατικό ρήμα και αντιστοιχεί σε *σχέση*. Ο τύπος οντότητας «λέξη» συνδέεται στη σχέση «διευκολύνει» ως υποκείμενο ενώ ο τύπος οντότητας «αναζήτηση» συνδέεται στη σχέση ως αντικείμενο. Το ER διάγραμμα που δημιουργείται από στην πρόταση φαίνεται παρακάτω.



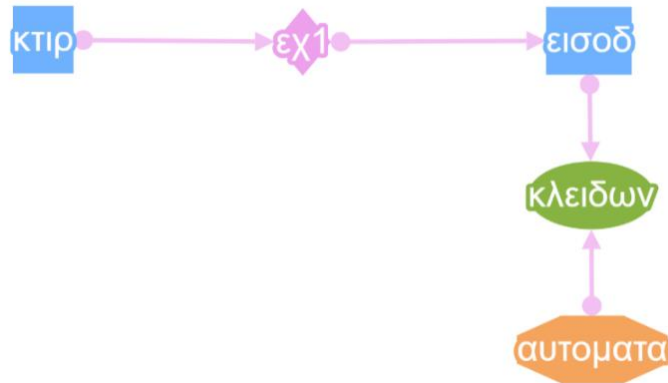
ΚΑΝΟΝΑΣ 10. Αν σε μία πρόταση περιέχεται η έκφραση «*X (του) Y*» με *X,Y ουσιαστικά* και το *Y* σε γενική πτώση, η έκφραση μετατρέπεται σε «*Y έχει X*».

Παράδειγμα

Πρόταση: «*Η είσοδος του κτιρίου κλειδώνει αυτόματα*».

Οι λέξεις «είσοδος» και «κτίριο» είναι ουσιαστικά, από τα οποία το δεύτερο σε πτώση γενική. Η φράση «*Η είσοδος του κτιρίου*» μετατρέπεται στην πρόταση «*Το κτίριο έχει είσοδο*». Το ER διάγραμμα

δημιουργείται από τις φράσεις «Το κτίριο έχει είσοδο» και «Η είσοδος κλειδώνει αυτόματα» σύμφωνα με τους προηγούμενους κανόνες όπως φαίνεται παρακάτω.



ΚΑΝΟΝΑΣ 11. Αν σε μία πρόταση περιέχεται η έκφραση «**X στο Y**» με **X,Y ουσιαστικά** και το **Y** σε αιτιατική πτώση, η έκφραση μετατρέπεται σε «**Y έχει X**».

Παράδειγμα

Πρόταση: «*Η υπογραφή στο έγγραφο δηλώνει την εγκυρότητα*»

Η λέξεις «υπογραφή» και «έγγραφο» είναι ουσιαστικά, από τα οποία το δεύτερο σε πτώση αιτιατική. Η φράση «*Η υπογραφή στο έγγραφο*» μετατρέπεται στην πρόταση «*Το έγγραφο έχει υπογραφή*». Το ER διάγραμμα δημιουργείται από τις φράσεις «*Το έγγραφο έχει υπογραφή*» και «*Η υπογραφή δηλώνει την εγκυρότητα*» σύμφωνα με τους προηγούμενους κανόνες όπως φαίνεται παρακάτω.



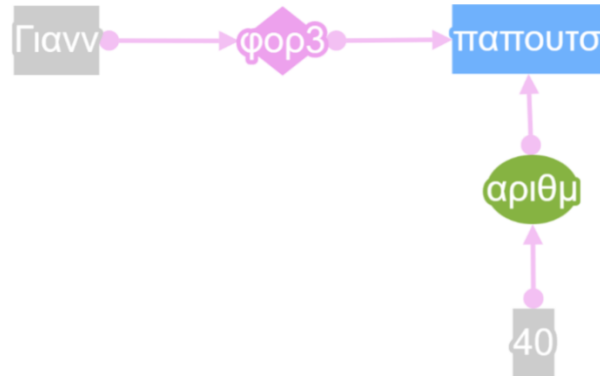
ΚΑΝΟΝΑΣ 12. Αν σε μία πρόταση περιέχεται η έκφραση «**X αριθμό|νούμερο|no Y**», το **X** είναι ουσιαστικό και το **Y** είναι **αριθμός**, τότε το **X** αντιστοιχεί σε **τύπο οντότητας** το οποίο έχει ως **ιδιότητα οντότητας** το «αριθμός» το οποίο έχει ως **ιδιότητα οντότητας** τον αριθμό **Y**.

Παράδειγμα

Πρόταση: «*Ο Γιάννης φοράει παπούτσια no 40*»

Στη φράση «*παπούτσια no 40*» το ουσιαστικό «παπούτσια» αντιστοιχεί σε **τύπο οντότητας**, το ουσιαστικό «*αριθμός*» εισάγεται ως **ιδιότητα οντότητας** και το αριθμητικό «*40*» αντιστοιχεί σε

οντότητα και συνδέεται ως *ιδιότητα οντότητας* στο «αριθμός». Το ER διάγραμμα που δημιουργείται σύμφωνα με τους προηγούμενους κανόνες φαίνεται παρακάτω.



Στον *Πίνακα 11* συνοψίζονται οι κανόνες δημιουργίας των στοιχείων του Μοντέλου Οντοτήτων – Συσχετίσεων από τη γραμματική των προτάσεων.

Κανόνας	Γραμματική	ER Model Rule
1	Ουσιαστικό (Common noun)	Τύπος Οντότητας (Entity Type)
2	Μεταβατικό ρήμα (Transitive Verb)	Σχέση (Relationship) μεταξύ οντοτήτων
3	Κύριο όνομα (Proper noun)	Οντότητα (Entity)
4	Επίθετο (Adjective)	Ιδιότητα οντότητας (Attribute of entity)
5	Επίρρημα (Adverb)	Ιδιότητα σχέσης (Attribute of relationship)
6	Αμετάβατο ρήμα (Intransitive verb)	Ιδιότητα οντότητας (Attribute of entity)
7	X είναι Y, X: Ουσιαστικό, Y: Επίθετο	Το X είναι τύπος οντότητας και το Y είναι Ιδιότητα Οντότητας
8	X Y, X: Ουσιαστικό, Y: Κύριο όνομα	Το Y είναι Οντότητα και το X είναι Ιδιότητα Οντότητας
	X Y, X: Ουσιαστικό, Y: Ουσιαστικό	Το X είναι Οντότητα και το Y είναι Ιδιότητα Οντότητας
9	X του Y ή X Y, X: Ουσιαστικό, Y: Ουσιαστικό στη γενική	Y έχει X
10	X στο Y, X: Ουσιαστικό, Y: Ουσιαστικό στην αιτιατική	Y έχει X
12	X αριθμό νούμερο νο Y, X: Ουσιαστικό, Y: αριθμός	Το X είναι Τύπος Οντότητας και το Y είναι Ιδιότητα Οντότητας

Πίνακας 11. Οι κανόνες δημιουργίας των στοιχείων του Μοντέλου Οντοτήτων – Συσχετίσεων από τη γραμματική των προτάσεων

5.3 Αποθήκευση της σημασιολογικής δομής αναπαράστασης ενός κειμένου στη βάση γνώσης

Η σημασιολογική αναπαράσταση ενός κειμένου αποθηκεύεται στη βάση γνώσης της Prolog μέσω του κατηγορήματος (ή της σχέσης) σημασιολογικής δομής, `relModel/8`, που περιέχει τις σημασιολογικές συνδέσεις όλων των στοιχείων του Μοντέλου Οντοτήτων – Συσχετίσεων (Τύπος Οντότητας, Οντότητα, Σχέση, Ιδιότητα Οντότητας, Ιδιότητα Σχέσης). Κάθε στοιχείο του μοντέλου καταχωρείται σε ένα από τα 5 κατηγορήματα (ή σε μια από τις 5 σχέσεις) ανάλογα με το τι αναπαριστά. Τα 5 κατηγορήματα των στοιχείων του ER Model που χρησιμοποιούνται είναι οι εξής:

1. Το κατηγορήμα "entity_types/3" αναπαριστά τους «τύπους οντοτήτων».
2. Το κατηγορήμα "entities/3" αναπαριστά τις «οντότητες».
3. Το κατηγορήμα "attributes_of_entities/3" αναπαριστά τις «ιδιότητες οντοτήτων».
4. Το κατηγορήμα "relationship_type/3" αναπαριστά τις «σχέσεις οντοτήτων».
5. Το κατηγορήμα "attributes_of_relationships/3" αναπαριστά τις «ιδιότητες των σχέσεων».

Όλα τα παραπάνω κατηγορήματα έχουν τα ίδια ορίσματα (Id,Name,LexType), τα οποία παριστούν τα εξής στοιχεία.

- Το **Id** είναι ο αύξοντας αριθμός του στοιχείου με αρχική τιμή το 1.
- Το **Name** είναι η ρίζα της λέξης.
- Το **LexType** παίρνει τις τιμές *noun, propernoun, verb, adjective, adverb*, ανάλογα με το μέρος του λόγου στο οποίο ανήκει η λέξη.

Στο κατηγορήμα (ή στη σχέση) αναπαράστασης της σημασιολογικής δομής *relModel/8* γίνεται αναφορά σε κάποιο από τα στοιχεία (Τύπος Οντότητας, Οντότητα, Σχέση, Ιδιότητα Οντότητας, Ιδιότητα Σχέσης) μέσω του ονόματος του αντίστοιχου κατηγορήματος και της ταυτότητας (Id) που έχει σ' αυτό το κατηγορήμα. Οι όροι «σχέση» και «κατηγορήμα» χρησιμοποιούνται εναλλακτικά διότι σε αυτή την αναπαράσταση της σημασιολογία των κειμένων χρησιμοποιούνται μόνο «πρότασεις γεγονότα». Κάθε «πρόταση γεγονός» μπορούμε να τη δούμε και ως «στιγμιότυπο μιας σχέσης». Στη συνέχεια θα χρησιμοποιούμε μόνο τη λέξη «κατηγορήμα» διότι η λέξη «σχέση» δημιουργεί σύγχυση με τις σχέσεις του ER model. Επιπλέον, για να μη υπάρχει σύγχυση με της χρήση της λέξης «πρόταση» όταν αναφερόμαστε σε κείμενο θα χρησιμοποιούμε τη λέξη «γεγονός» ή «πρόταση γεγονός» για να αναφερόμαστε στις προτάσεις ενός κατηγορήματος (ή στα στιγμιότυπα μιας σχέσης). Η αναπαράσταση των παραγόμενων σημασιολογικών δομών των οντοτήτων, των συσχετίσεων και των ιδιοτήτων τους γίνεται με το κατηγορήμα *relModel/8*.

relModel(Id,RefTable,RefId,ConnectedTo,ConnectedType,LexType,WordGrammar,MType)

Κάθε πρόταση γεγονός του κατηγορήματος *relModel/8* περιέχει ένα στοιχείο του ER Model (*entity, entity type, relationship, attribute of entity, attribute of relation*). Κάθε γεγονός μπορούμε να το δούμε και ως μια γραμμή ενός πίνακα. Τα ορίσματα του κατηγορήματος *relModel/8* παριστάνουν τα εξής στοιχεία της σημασιολογικής δομής του κειμένου.

- Το **Id** είναι ο αύξοντας αριθμός, με αρχική τιμή το 1, κάθε γεγονός του *relModel/8* που προστίθεται στη βάση γνώση και το οποίο αναπαριστά τμήμα της σημασιολογικής δομής του κειμένου το οποίο αντιστοιχεί στη σημασιολογία τμήματος του κειμένου.
- Το **RefTable** περιέχει μια τιμή δείκτη σε κάποιο γεγονός ενός εκ των κατηγορημάτων, *entity_types/3, relationship_type/3, attributes_of_entities/3, attributes_of_relationships/3* και *entities/3* στο οποίο γεγονός εμπεριέχεται το στοιχείο που καταχωρείται στο τρέχον γεγονός του *relModel/8* ανάλογα με τον τύπο του. Αν π.χ. το στοιχείο που καταχωρείται είναι *entity*, θα δωθεί η τιμή "entity", ενώ αν είναι *attribute of relationship* θα δωθεί η τιμή "relattr". Οι τιμές του RefTable ανάλογα με το κατηγορήμα στο οποίο γίνεται η αναφορά φαίνονται στον Πίνακα 12.

Τιμή RefTable	Κατηγορία Αναφοράς
relationship	relationship_type/3
entity	entity/3
attribute	attributes_of_entities/3
entityType	entity_types/3
relattr	attributes_of_relationships/3

Πίνακας 12. Οι τιμές του RefTable και το αντίστοιχο κατηγορία αναφοράς

- Το **RefId** περιέχει το Id του γεγονότος ενός από τα 5 προηγούμενα κατηγορήματα (στα οποία γίνεται αναφορά μέσω της τιμής του RefTable) στα οποία είναι καταχωρημένη η ρίζα και το μέρος του λόγου της λέξης που καταχωρείται στο τρέχον γεγονός. Δηλαδή αντί να καταχωρηθεί η ρίζα και το μέρος του λόγου της λέξης στο τρέχον γεγονός, καταχωρείται το Id με το οποίο γίνεται αναφορά στο γεγονός του κατηγορήματος στο οποίο περιέχονται.
- Το **ConnectedTo** περιέχει την τιμή του Id του γεγονότος με το οποίο συνδέεται το στοιχείο του τρέχοντος γεγονότος. Αν για παράδειγμα στο γεγονός με Id 3 καταχωρηθεί ένα ρήμα ως relationship και στο γεγονός με Id 4 ένα ουσιαστικό ως entity και το ουσιαστικό είναι το υποκείμενο του ρήματος στην πρόταση, τότε το ConnectedTo του γεγονότος με Id 4 θα έχει την τιμή του Id 3, το οποίο δηλώνει ότι το στοιχείο του γεγονότος με Id 4 συνδέεται με το στοιχείο του γεγονότος με Id 3. Αν το στοιχείο του τρέχοντος γεγονότος δε συνδέεται σε στοιχείο άλλου γεγονότος, το ConnectedTo έχει την τιμή 0.
- Το **ConnectedType** παίρνει τις τιμές 0 και 1. Η τιμή 0 δηλώνει ότι το στοιχείο του τρέχοντος γεγονότος είναι συνδεδεμένο στο στοιχείο του προηγούμενου γεγονότος (μέσω της τιμής του ConnectedTo) από αριστερά. **Αν** το ConnectedType έχει τιμή 1, **τότε** το στοιχείο του τρέχοντος γεγονότος είναι συνδεδεμένο στο στοιχείο του προηγούμενου γεγονότος από δεξιά. **Αν** το στοιχείο του τρέχοντος γεγονότος είναι ουσιαστικό καταχωρημένο ως entity και το στοιχείο του προηγούμενου γεγονότος στο οποίο συνδέεται είναι ένα μεταβατικό ρήμα καταχωρημένο ως relationship **τότε αν** η τιμή του ConnectedType είναι 0, το ουσιαστικό είναι συνδεδεμένο στο ρήμα ως υποκείμενο της πρότασης, **ενώ αν** η τιμή του ConnectedType είναι 1, το ουσιαστικό είναι συνδεδεμένο ως αντικείμενο της πρότασης. Επίσης **αν** ένα μεταβατικό ρήμα – relationship του τρέχοντος γεγονότος είναι συνδεδεμένο σε ένα ουσιαστικό – entity του προηγούμενου γεγονότος και το ConnectedType του ρήματος είναι 0, **τότε** το ρήμα συνδέεται αριστερά του ουσιαστικού, δηλαδή το ουσιαστικό είναι αντικείμενο της πρότασης. **Αν** αντίθετα το ConnectedType του ρήματος είναι 1, **τότε** το ρήμα συνδέεται δεξιά του ουσιαστικού, δηλαδή το ουσιαστικό είναι υποκείμενο της πρότασης.
- Το **LexType** παίρνει τις τιμές *noun, propernoun, verb, adjective, adverb*, ανάλογα με το μέρος του λόγου στο οποίο ανήκει η λέξη που καταχωρείται στο τρέχον γεγονός.
- Το **WordGrammar** είναι μια λίστα η οποία περιέχει τις πληροφορίες της γραμματικής της λέξης που καταχωρείται στο τρέχον γεγονός. Η λίστα είναι της μορφής

[Μέρος_του_λόγου,Ρίζα_Λέξης,..]. Ο αριθμός των στοιχείων της λίστας εξαρτάται από το μέρος του λόγου της λέξης.

- Το **Mtype** παίρνει τις τιμές **e** και **q**. Το **e** δηλώνει ότι η γραμμή που καταχωρείται είναι τμήμα μιας έκφρασης, ενώ το **q** ότι είναι τμήμα μιας ερώτησης.

Η δομή του κατηγορήματος σημασιολογικής δομής, *relModel/8*, παρέχει το πλεονέκτημα της αποθήκευσης και ανάκτησης πολύπλοκων δενδροειδών δομών με κόμβους και συνδέσεις μεταξύ τους. Κάθε κόμβος μπορεί να έχει από καμία έως απεριόριστες συνδέσεις με άλλους κόμβους ενώ δεν απαιτείται οι κόμβοι να εισαχθούν με προκαθορισμένη σειρά. Επιπλέον κάθε κόμβος περιέχει γραμματικές πληροφορίες για κάθε λέξη μαζί με το ρόλο της στο ER Model.

Παράδειγμα

Εισάγεται η φράση

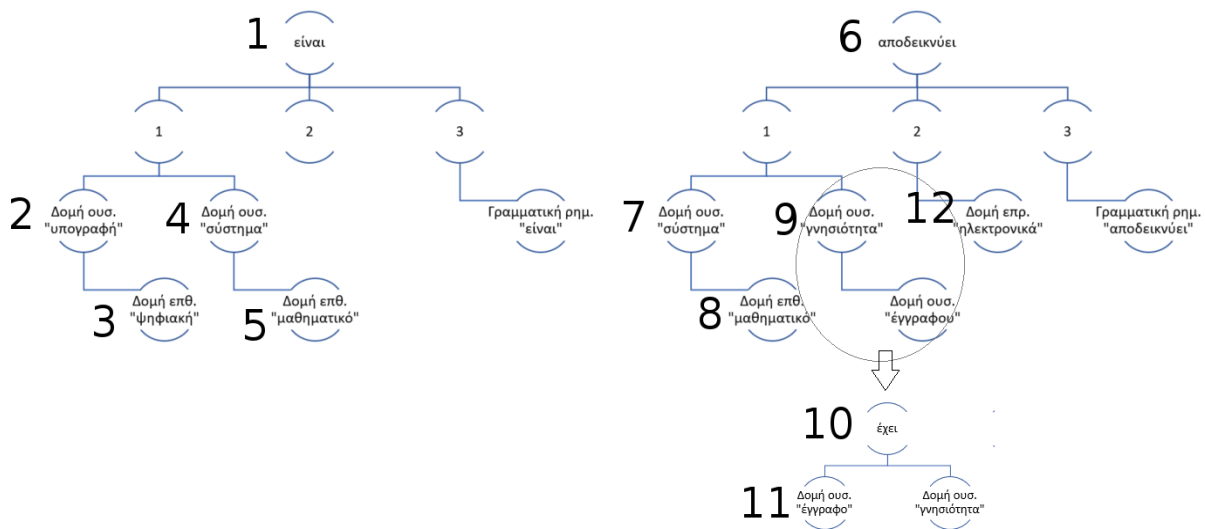
«Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά»

Η δομή της πρότασης που δημιουργείται μετά την επεξεργασία της φράσης είναι η ακόλουθη:

```
[ειν([υπογραφη([ψηφιακ([], [adjective,ψηφιακ,thiliko,enikos,onomastiki,1])),  
[noun,υπογραφη,thiliko,enikos,onomastiki,100])),  
[συστημ([μαθηματικ([], [adjective,μαθηματικ,ογdetero,enikos,aitiatiki,1])),  
[noun,συστημ,ογdetero,enikos,aitiatiki,203]]), [], [verb,ειν,trans,enikos,trito,10]),  
αποδεικνυ([συστημ([μαθηματικ([], [adjective,μαθηματικ,ογdetero,enikos,aitiatiki,1])),  
[noun,συστημ,ογdetero,enikos,aitiatiki,203])),  
γνησιοτητ(εγγραφ([], [noun,εγγραφ,ογdetero,enikos,geniki,200]),  
[noun,γνησιοτητ,thiliko,enikos,aitiatiki,6])),  
[ηλεκτρονικα([], [adverb,ηλεκτρονικα,tropiko])], [verb,αποδεικνυ,trans,enikos,trito,1]]]
```

Οι υπογραμμισμένες ρίζες των λέξεων αναφέρονται στα στοιχεία του ER Model τα οποία πρόκειται να καταχωρηθούν στο κατηγορήμα σημασιολογικής δομής.

Στο Σχήμα 25 που ακολουθεί φαίνεται η αριθμημένη σειρά των βημάτων (η οποία ταυτίζεται με το Id του κατηγορήματος σημασιολογικής δομής) για την καταχώρηση κάθε στοιχείου στο κατηγορήμα ακολουθώντας τους κανόνες του ER Model.



Σχήμα 25. Τα βήματα για την αποθήκευση στη βάση γνώση της αναπαράστασης της σημασιολογικής δομής της πρότασης «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά».

Βήμα 1:

Το πρώτο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του μεταβατικού ρήματος «είναι» το οποίο σύμφωνα με τον Κανόνα 2 του ER Model αντιστοιχεί σε relationship. Η ρίζα του «είναι» (ειν) και το μέρος του λόγου (verb) καταχωρούνται στο πρώτο γεγονός του κατηγορήματος relationship_type/3 με id=1, δηλαδή το πρώτο γεγονός του κατηγορήματος θα έχει τις τιμές (1,ειν,verb). Γνωρίζοντας το id του «είναι» και το όνομα του κατηγορήματος στο οποίο είναι καταχωρημένο, δημιουργείται το πρώτο γεγονός του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές:

Id=1,
 RefTable=relationship,
 RefId=1,
 ConnectedTo=0,
 ConnectedType=0,
 LexType=verb,
 WordGrammar=[verb,ειν,trans,enikos,trito,10],
 MType=e.

Δηλαδή οι τιμές του πρώτου γεγονότος του κατηγορήματος relModel/8 είναι

relModel(1,relationship,1,0,0,verb,[verb,ειν,trans,enikos,trito,10],e)

Υπενθυμίζεται ότι τα ορίσματα του relModel/8 έχουν την ακόλουθη σειρά:

relModel (Id, RefTable, RefId, ConnectedTo, ConnectedType, LexType, WordGrammar, MType)

Το ConnectedTo έχει τιμή 0 γιατί το ρήμα «είναι» δε συνδέεται σε κάποιο άλλο γεγονός του κατηγορήματος relModel/8 .

Βήμα 2:

Το δεύτερο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του ουσιαστικού «υπογραφή» το οποίο συνδέεται στο «είναι» εφόσον η δομή του είναι στη λίστα των παραμέτρων της δομής του «είναι». Σύμφωνα με τον Κανόνα 1 του ER Model, ως ουσιαστικό αντιστοιχεί σε entity. Η ρίζα του «υπογραφή» (υπογραφ) και το μέρος του λόγου (noun) καταχωρούνται στο πρώτο γεγονός του κατηγορήματος entity_types με id=1, δηλαδή το πρώτο γεγονός του κατηγορήματος θα έχει τις τιμές (1,υπογραφ,noun). Γνωρίζοντας το id του «υπογραφή» και το όνομα του κατηγορήματος στο οποίο είναι καταχωρημένο, δημιουργείται το δεύτερο γεγονός του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές :

Id=2,
RefTable=entity,
RefId=1,
ConnectedTo=1,
ConnectedType=0,
LexType=noun,
WordGrammar =[noun,υπογραφ,thiliko,enikos,onomastiki,100],
MType=e.

Δηλαδή οι τιμές του δεύτερου γεγονότος του κατηγορήματος relModel/8 είναι

relModel(2,entity,1,1,0, noun, [noun,υπογραφ,thiliko,enikos,onomastiki,100],e)

Βήμα 3:

Το τρίτο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του επιθέτου «ψηφιακή» το οποίο συνδέεται στο «υπογραφή» εφόσον η δομή του είναι στη λίστα των παραμέτρων της δομής του «υπογραφή». Σύμφωνα με τον Κανόνα 4 του ER Model, ως επίθετο αντιστοιχεί σε attribute of an entity. Η ρίζα του «ψηφιακή» (ψηφιακ) και το μέρος του λόγου (adjective) καταχωρούνται στο πρώτο γεγονός του κατηγορήματος attributes_of_entities με id=1, δηλαδή το πρώτο γεγονός του κατηγορήματος θα έχει τις τιμές (1, ψηφιακ, adjective). Γνωρίζοντας το id του «ψηφιακή» και το όνομα του κατηγορήματος στον οποίο είναι γεγονός, δημιουργείται το τρίτο γεγονός του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές:

Id=3,
RefTable=attributes_of_entities,
RefId=1,
ConnectedTo=2,
ConnectedType=0,

LexType= adjective,
WordGrammar=[adjective,ψηφιακ,thiliko,enikos,onomastiki,1],
MType=e.

Δηλαδή οι τιμές του τρίτου γεγονότος του κατηγορήματος relModel/8 είναι

relModel(3,attribute,1,2,0,adjective,[adjective,ψηφιακ,thiliko,enikos,onomastiki,1],e)

Βήμα 4:

Το τέταρτο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του ουσιαστικού «σύστημα» το οποίο συνδέεται στο «είναι» εφόσον η δομή του είναι στη λίστα των παραμέτρων της δομής του «είναι» και μάλιστα ως αντικείμενο της πρότασης. Σύμφωνα με τον Κανόνα 1 του ER Model, ως ουσιαστικό αντιστοιχεί σε entity. Η ρίζα του «σύστημα» (σύστημ) και το μέρος του λόγου (noun) καταχωρούνται στο δεύτερο γεγονός του κατηγορήματος entity_types/3 με id=2, δηλαδή το δεύτερο γεγονός του κατηγορήματος θα έχει τις τιμές (2, συστημ, noun). Γνωρίζοντας το id του «σύστημα» και το όνομα του πίνακα στον οποίο είναι καταχωρημένο, δημιουργείται το τέταρτο γεγονός του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές:

Id=4,
RefTable=entity,
RefId=2,
ConnectedTo=1,
ConnectedType=1,
LexType=noun,
WordGrammar=[noun,συστημ,oydetero,enikos,aitiatiki,203]
MType=e.

Δηλαδή οι τιμές του τέταρτου γεγονότος του κατηγορήματος relModel/8 είναι

relModel(4,entity,2,1,1,noun,[noun,συστημ,oydetero,enikos,aitiatiki,203],e)

Σ' αυτήν την καταχώρηση, σε αντίθεση με όλες τις προηγούμενες, το ConnectedType έχει τιμή 1 γιατί το ουσιαστικό «σύστημα» συνδέεται στο ρήμα «είναι» στα δεξιά του, δηλαδή ως αντικείμενο.

Βήμα 5:

Το πέμπτο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του επιθέτου «μαθηματικό» το οποίο συνδέεται στο «σύστημα» εφόσον η δομή του είναι στη λίστα των παραμέτρων της δομής του «μαθηματικό». Σύμφωνα με τον Κανόνα 4 του ER Model, ως επίθετο αντιστοιχεί σε attribute of an entity. Η ρίζα του «μαθηματικό» (μαθηματικ) και το μέρος του λόγου (adjective) καταχωρούνται στο δεύτερο γεγονός του κατηγορήματος attributes_of_entities/3 με id=2, δηλαδή το δεύτερο γεγονός του κατηγορήματος θα έχει τις τιμές (2, μαθηματικ, adjective). Γνωρίζοντας το id του «μαθηματικό» και το όνομα του κατηγορήματος στον οποίο ανήκει το γεγονός, δημιουργείται το πέμπτο γεγονός του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές:

Id=5,
RefTable=attributes_of_entities,
RefId=2,
ConnectedTo=4,
ConnectedType=0,
LexType= adjective,
WordGrammar=[adjective,μαθηματικ,ογdetero,enikos,aitiatici,1],
MType=e.

Δηλαδή οι τιμές του πέμπτου γεγονότος του κατηγορήματος relModel/8 είναι

relModel(5, attribute,2,4,0,adjective,[adjective,μαθηματικ,ογdetero,enikos,aitiatici,1],e)

Βήμα 6:

Το έκτο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του μεταβατικού ρήματος «αποδεικνύει» το οποίο σύμφωνα με τον Κανόνα 2 του ER Model αντιστοιχεί σε relationship. Η ρίζα του «αποδεικνύει» (αποδεικνυ) και το μέρος του λόγου (verb) καταχωρούνται στο δεύτερο γεγονός του κατηγορήματος relationship_type/3 με id=2, δηλαδή το δεύτερο γεγονός του κατηγορήματος θα έχει τις τιμές (2, αποδεικνυ, verb). Γνωρίζοντας το id του «αποδεικνυει» και το όνομα του κατηγορήματος στον οποίο ανήκει το γεγονός, δημιουργείται το έκτο γεγονός του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές:

Id=6,
RefTable=relationship,
RefId=2,
ConnectedTo=0,
ConnectedType=0,
LexType=verb,
WordGrammar=[verb,αποδεικνυ,trans,enikos,trito,1]
MType=e.

Δηλαδή οι τιμές του έκτου γεγονότος του κατηγορήματος relModel/8 είναι

relModel(6,relationship,2,0,0,verb,[verb,αποδεικνυ,trans,enikos,trito,1],e)

Το ConnectedTo έχει τιμή 0 γιατί το ρήμα «αποδεικνύει» δε συνδέεται σε κάποιο άλλο στοιχείο του πίνακα.

Βήμα 7:

Το έβδομο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του ουσιαστικού «σύστημα» το οποίο συνδέεται στο «αποδεικνύει» εφόσον η δομή του είναι στη λίστα των παραμέτρων της δομής του «αποδεικνύει». Σύμφωνα με τον Κανόνα 1 του ER Model, ως ουσιαστικό αντιστοιχεί σε entity. Η δομή του «σύστημα» είναι ήδη καταχωρημένη στο δεύτερο γεγονός του κατηγορήματος

entity_types/3 με id=2 και δεν καταχωρείται ξανά. Με αυτό το id και το όνομα του κατηγορήματος στο οποίο ανήκει το γεγονός, δημιουργείται το έβδομο γεγονός του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές:

```
Id=7,  
RefTable=entity,  
RefId=2,  
ConnectedTo=6,  
ConnectedType=0,  
LexType=noun,  
WordGrammar=[noun,συστημ,ογdetero,enikos,aitiatiki,203],  
MType=e.
```

Δηλαδή οι τιμές του έβδομου γεγονότος του κατηγορήματος relModel/8 είναι

```
relModel(7,entity,2,6,0, noun,[noun,συστημ,ογdetero,enikos,aitiatiki,203],e)
```

Βήμα 8:

Το όγδοο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του επιθέτου «μαθηματικό» το οποίο συνδέεται στο «σύστημα» εφόσον η δομή του είναι στη λίστα των παραμέτρων της δομής του «σύστημα». Σύμφωνα με τον Κανόνα 4 του ER Model, ως επίθετο αντιστοιχεί σε attribute of an entity. Η δομή του «μαθηματικό» είναι ήδη καταχωρημένη στο δεύτερο γεγονός του κατηγορήματος attributes_of_entities/3 με id=2 και δεν καταχωρείται ξανά. Γνωρίζοντας το id του «μαθηματικό» και το όνομα του κατηγορήματος στο οποίο ανήκει το γεγονός, δημιουργείται το όγδοο γεγονός του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές:

```
Id=8,  
RefTable=attributes_of_entities,  
RefId=2,  
ConnectedTo=7,  
ConnectedType=0,  
LexType= adjective,  
WordGrammar=[adjective,μαθηματικ,ογdetero,enikos,aitiatiki,1],  
MType=e.
```

Δηλαδή οι τιμές του όγδοου γεγονότος του κατηγορήματος relModel/8 είναι

```
relModel(8,attributes_of_entities,2,7,0,adjective,[adjective,μαθηματικ,ογdetero,enikos,aitiatiki,1],e)
```

```
relModel (Id, RefTable, RefId, ConnectedTo, ConnectedType, LexType, WordGrammar, MType  
e)
```

Βήμα 9:

Το ένατο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του ουσιαστικού «γνησιότητα» το οποίο συνδέεται στο «αποδεικνύει» εφόσον η δομή του είναι στη λίστα των παραμέτρων της δομής του «αποδεικνύει» και μάλιστα ως αντικείμενο της πρότασης. Σύμφωνα με τον Κανόνα 1 του ER Model, ως ουσιαστικό αντιστοιχεί σε entity. Η ρίζα του «γνησιότητα» (γνησιότητ) και το μέρος του λόγου (noun) καταχωρούνται στο τρίτο γεγονός του κατηγορήματος entity_types/3 με id=3, δηλαδή το τρίτο γεγονός του κατηγορήματος θα έχει τις τιμές (3, γνησιότητ, noun). Με αυτό το id και το όνομα του κατηγορήματος στο οποίο ανήκει το γεγονός, δημιουργείται το ένατο γεγονός του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές:

Id=9,
RefTable=entity,
RefId=3,
ConnectedTo=6,
ConnectedType=0,
LexType=noun,
WordGrammar=[noun,γνησιότητ,thiliko,enikos,aitiatiki,6],
MType=e.

Δηλαδή οι τιμές του ένατου γεγονότος του κατηγορήματος relModel/8 είναι

relModel(9,entity,3,6,0, noun,[noun,γνησιότητ,thiliko,enikos,aitiatiki,6],e)

Βήμα 10:

Το επόμενο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του ουσιαστικού «εγγράφου» το οποίο συνδέεται στο «γνησιότητα» εφόσον η δομή του είναι στη λίστα των παραμέτρων της δομής του «γνησιότητα». Δηλαδή η δομή του ουσιαστικού «εγγράφου», βρίσκεται στη λίστα των παραμέτρων του ουσιαστικού «γνησιότητα». Σύμφωνα με τον Κανόνα 9 του ER Model, η φράση

«γνησιότητα εγγράφου»

Θα μετατραπεί στη φράση

«το έγγραφο έχει γνησιότητα»

Η δομή του ουσιαστικού «γνησιότητα» έχει ήδη καταχωρηθεί από το προηγούμενο βήμα, οπότε μένει να καταχωρηθούν οι δομές του ρήματος «έχει» και του ουσιαστικού «έγγραφο».

Η δομή του μεταβατικού ρήματος «έχει» είναι το δέκατο γεγονός του κατηγορήματος relModel/8 το οποίο σύμφωνα με τον Κανόνα 2 του ER Model αντιστοιχεί σε relationship. Η ρίζα του «έχει» (έχ) και το μέρος του λόγου (verb) καταχωρούνται στο τρίτο γεγονός του κατηγορήματος relationship_type/3 με id=3, δηλαδή το τρίτο γεγονός του κατηγορήματος θα έχει τις τιμές (3, έχ, verb). Γνωρίζοντας το id του «έχει» και το όνομα του κατηγορήματος στο οποίο ανήκει το γεγονός, δημιουργείται το δέκατο γεγονός

του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές:

Id=10,
RefTable=relationship,
RefId=3,
ConnectedTo=9,
ConnectedType=0,
LexType=verb,
WordGrammar=[verb,εχ,trans,enikos,trito,1]
MType=e.

Δηλαδή οι τιμές του δέκατου γεγονότος του κατηγορήματος relModel/8 είναι

relModel(10,relationship,3,9,0,verb,[verb,εχ,trans,enikos,trito,1],e)

relModel (Id, RefTable, RefId, ConnectedTo, ConnectedType, LexType, WordGrammar, MType
e)

Η δομή του ρήματος «έχει» συνδέεται στο γεγονός με ταυτότητα 9 με τη δομή του ουσιαστικού «γνησιότητα» με connectedType=0, δηλαδή το «έχει» συνδέεται αριστερά του «γνησιότητα» το οποίο είναι αντικείμενο της πρότασης.

Βήμα 11:

Το ενδέκατο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του ουσιαστικού «εγγράφου» του οποίου η αρχική σύνδεση ήταν με το ουσιαστικό «γνησιότητα», αλλά λόγω της εφαρμογής στο προηγούμενο βήμα του κανόνα 9 του ER Model, θα συνδεθεί στο ρήμα «έχει». Σύμφωνα με τον Κανόνα 1 του ER Model, ως ουσιαστικό αντιστοιχεί σε entity. Η ρίζα του «εγγράφου» (εγγραφ) και το μέρος του λόγου (noun) καταχωρούνται στο τέταρτο γεγονός του κατηγορήματος entity_types/3 με id=4, δηλαδή το τέταρτο γεγονός του κατηγορήματος θα έχει τις τιμές (4,εγγραφ,noun). Με αυτό το id και το όνομα του κατηγορήματος στο οποίο ανήκει το γεγονός, δημιουργείται το ενδέκατο γεγονός του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές:

Id=11,
RefTable=entity,
RefId=4,
ConnectedTo=10,
ConnectedType=0,
LexType=noun,
WordGrammar=[noun,εγγραφ,oydetero,enikos,geniki,200],
MType=e.

Δηλαδή οι τιμές του ενδέκατου γεγονότος του κατηγορήματος relModel/8 είναι

relModel(11,entity,4,10,0, noun,[noun,εγγραφ,oydetero,enikos,geniki,200], e)

relModel (Id, RefTable, RefId, ConnectedTo, ConnectedType, LexType, WordGrammar, MType)

Η δομή του ουσιαστικού «έγγραφο» συνδέεται στο γεγονός με ταυτότητα 10 με τη δομή του ρήματος «έχει» με connectedType=0, δηλαδή το «έγγραφο» συνδέεται αριστερά του «έχει» συνδέεται δηλαδή ως υποκείμενο της πρότασης.

Βήμα 12:

Το τελευταίο στοιχείο της σημασιολογικής δομής της πρότασης είναι αυτό του επιρρήματος «ηλεκτρονικά» το οποίο συνδέεται στο «αποδεικνύει» εφόσον η δομή του είναι στη λίστα των παραμέτρων της δομής του «αποδεικνύει». Σύμφωνα με τον Κανόνα 5 του ER Model, ως επίρρημα αντιστοιχεί σε attribute of relationship. Η ρίζα του «ηλεκτρονικά» (ηλεκτρονικ) και το μέρος του λόγου (adverb) καταχωρούνται στο πρώτο γεγονός του κατηγορήματος attributes_of_relationships/3 με id=1, δηλαδή το πρώτο γεγονός του κατηγορήματος θα έχει τις τιμές (1, ηλεκτρονικ, adverb). Γνωρίζοντας το id του «ηλεκτρονικά» και το όνομα του κατηγορήματος στο οποίο ανήκει το γεγονός, δημιουργείται το δωδέκατο γεγονός του κατηγορήματος σημασιολογικής αναπαράστασης του κειμένου relModel/8 από τις παρακάτω τιμές:

Id=12,
RefTable=relattr,
RefId=1,
ConnectedTo=6,
ConnectedType=0,
LexType=adverb,
WordGrammar=[adverb,ηλεκτρονικα,tropiko],
MType=e.

Δηλαδή οι τιμές του δωδέκατου γεγονότος του κατηγορήματος relModel/8 είναι

relModel(12,adverb,1,6,0,noun,[adverb,ηλεκτρονικα,tropiko],e)

Μετά το τέλος της διαδικασίας αυτόματης δημιουργίας της σημασιολογικής αναπαράστασης του κειμένου του παραδείγματος, όλα τα γεγονότα του κατηγορήματος relModel/8 παρουσιάζονται για διευκόλυνση στη κατανόηση τους από τον αναγνώστη σε μορφή πίνακα, Πίνακα 13. Επιπλέον, η 4^η στήλη (Ρίζα) η οποία δείχνει σε ποια λέξη αναφέρεται το αντίστοιχο γεγονός, δεν υπάρχει ως όρισμα στο κατηγορήμα relModel/8, καθώς υπολογίζεται από τη γνώση των ορισμάτων RefTable και RefId αλλά έχει προστεθεί για καλύτερη κατανόηση της σημασιολογίας κάθε γεγονότος. Με τον ίδιο αριθμό στηλών εμφανίζονται και οι επόμενοι πίνακες του relModel/8.

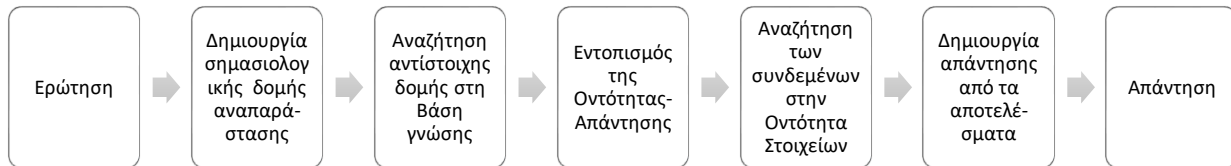
Id	RefTable	RefId	(Ρίζα)	ConectedTo	Connect edType	LexType	WordGrammar	MType
1	relationship	4	ειν	0	0	verb	[verb,ειν,trans,enikos,trito,10]	e
2	entity	1	υπογραφ	1	0	noun	[noun,υπογραφ,thiliko,enikos,onomastiki,100]	e
3	attribute	1	ψηφιακ	2	0	adjective	[adjective,ψηφιακ,thiliko,enikos,onomastiki,1]	e
4	entity	2	συστημ	1	1	noun	[noun,συστημ,oydetero,enikos,aitiatiiki,203]	e
5	attribute	2	μαθηματικ	4	0	adjective	[adjective,μαθηματικ,oydetero,enikos,aitiatiiki,1]	e
6	relationship	5	αποδεικνυ	0	0	verb	[verb,αποδεικνυ,trans,enikos,trito,1]	e
7	entity	2	συστημ	6	0	noun	[noun,συστημ,oydetero,enikos,aitiatiiki,203]	e
8	attribute	2	μαθηματικ	7	0	adjective	[adjective,μαθηματικ,oydetero,enikos,aitiatiiki,1]	e
9	entity	3	γνησιοτητ	6	1	noun	[noun,γνησιοτητ,thiliko,enikos,aitiatiiki,6]	e
10	relationship	6	εχ	9	0	verb	[verb,εχ,trans,enikos,trito,1]	e
11	entity	4	εγγραφ	10	0	noun	[noun,εγγραφ,oydetero,enikos,geniki,200]	e
12	relattr	1	ηλεκτρονικα	6	0	adverb	[adverb,ηλεκτρονικα,tropiko]	e

Πίνακας 13. Τα γεγονότα του κατηγορήματος relModel/8 σε μορφή πίνακα τα οποία παριστούν τη σημασιολογική δομή του κειμένου του παραδείγματος

5.4 Απαντήσεις σε ερωτήματα

5.4.1 Περιγραφή

Η εφαρμογή παρέχει τη δυνατότητα σύνθεσης απαντήσεων σε φυσική γλώσσα σε ερωτήματα της μορφής «ποιος» και «τι» τα οποία εισάγονται ως κείμενο από το χρήστη, αξιοποιώντας τη σημασιολογική αναπαράσταση των κειμένων που υπάρχουν στη βάση γνώση του συστήματος μας. Η αρχιτεκτονική του συστήματος δημιουργίας των απαντήσεων στα ερωτήματα των χρηστών φαίνεται στο Σχήμα 26.



Σχήμα 26: Η αρχιτεκτονική του συστήματος δημιουργίας απαντήσεων σε ερωτήσεις

Οι ερωτήσεις αρχικά επεξεργάζονται με τον ίδιο τρόπο που επεξεργάζονται οι προτάσεις. Δηλαδή, με βάση το κείμενο κάθε ερώτησης δημιουργείται σύμφωνα με το ER Model η σημασιολογική δομή του η οποία στη συνέχεια αναπαρίσταται στη βάση γνώσης ως γεγονότα του κατηγορήματος relModel/8. Στη συνέχεια καθορίζεται ο τύπος της απάντησης που θα αναζητηθεί ανάλογα με την ερωτηματική αντωνυμία της ερώτησης και μετά εντοπίζεται το ρήμα της ερώτησης. Ξεκινώντας από το ρήμα ως σχέση (relation) μιας δομής, γίνεται αναζήτηση των συνδεδεμένων σε αυτό στοιχείων της δομής της ερώτησης (entities, attributes) στον πίνακα της βάσης. Αν η ερωτηματική αντωνυμία είναι το «**ποιος**», αναζητείται το υποκείμενο που είναι συνδεδεμένο στο ρήμα, ενώ αν η ερωτηματική αντωνυμία είναι το «**τι**», αναζητείται το αντικείμενο. Για τις ερωτήσεις τύπου «**τι**»/«**ποιος**» η απάντηση είναι τύπου entity ή entity type. Εφόσον εντοπιστεί το entity ή το entity type, αναζητούνται τα attributes και entities με τα οποία είναι συνδεδεμένο. Τα αποτελέσματα αναζήτησης χρησιμοποιούνται για τη δημιουργία του αρχικού τμήματος του κειμένου της απάντησης αφού εφαρμοστούν οι κατάλληλοι γραμματικοί και συντακτικοί κανόνες. Το τελικό τμήμα της απάντησης βασίζεται στη διατύπωση της ερώτησης.

5.4.2 Αλγόριθμος δημιουργίας απάντησης

Τα κείμενα που εισάγονται στην εφαρμογή θεωρούνται ερωτήσεις εφόσον ξεκινούν με τις ερωτηματικές αντωνυμίες «ποιος» και «τι» και τελειώνουν με ερωτηματικό. Για κάθε ερώτηση δημιουργείται μια σημασιολογική δομή αναπαράστασης με βάση το ER Model και αναπαρίσταται στη βάση γνώσης ως γεγονότα του κατηγορήματος relModel/8. Δηλαδή, οι ερωτήσεις επεξεργάζονται και αποθηκεύονται όπως και οι προτάσεις ενός κειμένου, αλλά με τις εξής διαφορές:

1. Ο τύπος της ερωτηματικής αντωνυμίας στο ER Model αναφέρεται ως “question” στο όρισμα RefTable.
2. Το Mtype στον πίνακα relModel έχει την τιμή “q” που δηλώνει ότι πρόκειται για ερώτηση.

3. Τα γεγονότα του relModel/8 με Mtype=q έχουν χωριστή αρίθμηση ταυτότητας (Id), δηλαδή η αρίθμηση του Id στο relModel/8 με Mtype=q ξεκινάει από 1.

Παράδειγμα: Αν το κατηγορήμα αναπαράστασης της σημασιολογικής δομής relModel/8 δεν έχει κανένα γεγονός και εισαχθεί το κείμενο «Η δασκάλα διδάσκει ιστορία» θα δημιουργηθούν 3 γεγονότα με την αναπαράσταση της σημασιολογικής δομής της πρότασης. Για καλύτερη επίδειξη των τιμών των ορισμάτων τους επιδεικνύονται με μορφή πίνακα, Πίνακα 14.

Id	RefTable	RefId	(Ρίζα)	ConectedTo	Connect edType	LexType	WordGrammar	MTy pe
1	relationship	1	διδασκ	0	0	verb	[verb,διδασκ,trans,enikos,trito,1]	e
2	entity	1	δασκαλ	1	0	noun	[noun,δασκαλ,thiliko,enikos,onomastiki,6]	e
3	entity	2	ιστορ	1	1	noun	[noun,ιστορ,thiliko,enikos,aitiatiki,8]	e

Πίνακας 14. Πίνακας με τις τιμές των ορισμάτων τριών γεγονότων του κατηγορήματος relModel/3 τα οποία δημιουργούνται με την αναπαράσταση της σημασιολογικής δομής της πρότασης «Η δασκάλα διδάσκει ιστορία»

Στη συνέχεια εισάγεται η ερώτηση «Ποιος διδάσκει ιστορία;». Η σημασιολογική αναπαράσταση της δομής της ερώτησης θα δημιουργήσει 3 γεγονότα του relModel/8. Όλα τα γεγονότα του κατηγορήματος relModel/8 για καλύτερη επίδειξη των τιμών των ορισμάτων τους επιδεικνύονται με μορφή πίνακα στον Πίνακα 15.

Id	RefTable	RefId	(Ρίζα)	ConectedTo	Connect edType	LexType	WordGrammar	MTy pe
1	relationship	1	διδασκ	0	0	verb	[verb,διδασκ,trans,enikos,trito,1]	e
2	entity	1	δασκαλ	1	0	noun	[noun,δασκαλ,thiliko,enikos,onomastiki,6]	e
3	entity	2	ιστορ	1	1	noun	[noun,ιστορ,thiliko,enikos,aitiatiki,8]	e
1	relationship	1	διδασκ	0	0	verb	[verb,διδασκ,trans,enikos,trito,1]	q
2	question	3	ποιος	1	0	er_ antwnymia	[er_antwnymia,ποιος,arseniko,enikos,onomastiki]	q
3	entity	2	ιστορ	1	1	noun	[noun,ιστορ,thiliko,enikos,aitiatiki,8]	q

Πίνακας 15. Πίνακας με τις τιμές των ορισμάτων όλων των γεγονότων του κατηγορήματος relModel/3 τα οποία δημιουργούνται με την αναπαράσταση της σημασιολογικής δομής της πρότασης «Η δασκάλα διδάσκει ιστορία» και της ερώτησης «Ποιος διδάσκει ιστορία;»

Στο στάδιο αυτό έχει γίνει η αναπαράσταση τόσο του κειμένου όσο και της ερώτησης και οι δομές τους έχουν αποθηκευτεί ως γεγονότα του relModel/8. Η ταυτότητα κάθε γεγονότος αποτελείται από το ζεύγος (Id, MType). Το επόμενο στάδιο αφορά στη σύγκριση των σημασιολογικών δομών για την εξαγωγή της απάντησης.

Για τον προσδιορισμό και τη σύνθεση της απάντησης ακολουθούνται τα παρακάτω βήματα:

1. Βρες την ερωτηματική αντωνυμία («ποιος», «τι») στο γεγονός του relModel/8 της βάσης γνώσης αναζητώντας γεγονότα με τιμή «q» στο όρισμα Mtype, δηλαδή Mtype=q (δομή ερώτησης).

2. Βρες το ρήμα V που είναι συνδεδεμένο με την ερωτηματική αντωνυμία στο γεγονός με τιμή « q » στο όρισμα $Mtype$, $Mtype=q$ (δομή ερώτησης).
3. Βρες το ρήμα V στα γεγονότα του $relModel/8$ με τιμή « e » στο όρισμα $Mtype$, $Mtype=e$ (δομές προτάσεων).
4. Εντόπισε το ID του γεγονότος του ρήματος V που βρήκες.
5. Δημιούργησε τη λίστα $ConnectedIDs$ των IDs που είναι συνδεδεμένα στο προηγούμενο.
6. Χρησιμοποιώντας το ID και τη λίστα $ConnectedIDs$ δημιούργησε το κείμενο του υποκειμένου ή του αντικειμένου (ανάλογα με τον τύπο της ερώτησης, «ποιος» ή «τι») της απάντησης.
7. Χρησιμοποιώντας το προηγούμενο κείμενο, το ρήμα και το κείμενο της ερώτησης, κατασκεύασε την πλήρη πρόταση της απάντησης με δομή υποκείμενο-ρήμα-αντικείμενο.

Για την εκτέλεσή τους έχει υλοποιηθεί στην Prolog ο παρακάτω αλγόριθμος:

Έστω QT η ακολουθία λέξεων του κειμένου της ερώτησης. Το QT έχει τη μορφή [Ποιος/Τι, Ρήμα, ΥπόλοιποΚείμενοQT], δηλαδή $QT := [Ποιος/Τι, Ρήμα, ΥπόλοιποΚείμενοQT]$.

Έστω το σύνολο $Q=\{(r,q): r \text{ είναι η τιμή του ορίσματος } Id \text{ του κατηγορήματος } RelModel/8, \text{ δηλαδή } Id=r, \text{ και } q \text{ είναι η τιμή του ορίσματος } Mtype, \text{ δηλαδή } Mtype=q\}$

Βρες από το Q το στοιχείο (r,q) για το οποίο στο γεγονός με ταυτότητα (r,q) και στο πεδίο $WordGrammar$ υπάρχει η ερωτηματική αντωνυμία («ποιος», «τι»). Έστω QW η ερωτηματική αντωνυμία.

Έστω r' η τιμή του ορίσματος $ConnectedTo$ του γεγονότος (r,q) του $RelModel/8$. Βρες από το γεγονός (r',q) και το όρισμα $WordGrammar$ του $RelModel/8$ το ρήμα V και την ταυτότητα Vid .

Έστω $E=\{(r,e): r \text{ είναι η τιμή του ορίσματος } Id \text{ ενός γεγονότος του } RelModel, \text{ δηλαδή } Id=r, \text{ και } e \text{ είναι η τιμή του πεδίου } Mtype, \text{ δηλαδή } Mtype=e\}$.

Βρες από το E το στοιχείο (r'',e) για το οποίο στο γεγονός (r'',e) η τιμή στο όρισμα $RefId$ είναι Vid

Έστω σύνολο $C=\{r_c: r_c \text{ είναι η τιμή του ορίσματος } Id \text{ για ένα γεγονός } (r,e) \in E \text{ τέτοια ώστε η τιμή του ορίσματος } ConnectedTo \text{ να είναι } r''\}$

Εάν $QW=\text{"ποιος"}$ τότε

Υποκείμενο := ΦτιάξεΥποκείμενο("ποιος", r'' , C);

Απάντηση := ΦτιάξεΑπάντηση(Vid , Υποκείμενο, ΥπόλοιποΚείμενοQT);

Τέλος_Εάν

Εάν $QW=\text{"τι"}$ τότε

Αντικείμενο:=ΦτιάξεΑντικείμενο("τι", r'' , C);

Απάντηση := ΦτιάξεΑπάντηση(Vid , Αντικείμενο, ΥπόλοιποΚείμενοQT);

Τέλος_Εάν

Στο προηγούμενο παράδειγμα ο χρήστης εισήγαγε την ερώτηση «Ποιος διδάσκει ιστορία». Σύμφωνα με τον αλγόριθμο, πρώτα θα αναζητηθεί η ερωτηματική αντωνυμία που είναι το «Ποιος» και θα καθοριστεί ότι αναζητείται το υποκείμενο της πρότασης (γεγονός του RelModel/8 με id=1 και Mtype=q). Στη συνέχεια θα αναζητηθεί το ρήμα της πρότασης που είναι το «διδάσκει» (γεγονός του RelModel/8 με id=2 και Mtype=q). Ακολούθως θα γίνει αναζήτηση του «διδάσκει» ως relationship στη σημασιολογική αναπαράσταση της πρότασης (γεγονός του RelModel/8 με id=1 και Mtype=e). Εφόσον εντοπιστεί το relationship θα αναζητηθεί το entity ή το entity type που είναι συνδεδεμένο μαζί του ως υποκείμενο και θα εντοπιστεί το «δασκάλα» (γεγονός του RelModel/8 με id=2, Mtype=e). Θα δημιουργηθεί η φράση «Η δασκάλα» η οποία ακολουθείται από το ρήμα της ερώτησης «διδάσκει», το οποίο με τη σειρά του ακολουθείται από το υπόλοιπο κείμενο της ερώτησης δηλαδή «ιστορία». Από τα τρία αυτά συνθετικά θα δημιουργηθεί η πρόταση της απάντησης: «Η δασκάλα διδάσκει ιστορία».

Σε περίπτωση που η αναζήτηση του ρήματος της ερώτησης στο κείμενο αποτύχει, ή αν βρεθεί το ρήμα αλλά η υπόλοιπη δομή δεν ταυτίζεται με αυτήν της ερώτησης, η εφαρμογή δεν είναι σε θέση να απαντήσει στην ερώτηση και επιστρέφει τη φράση «Λυπάμαι. Δε γνωρίζω».

Εξάιρεση αποτελεί η περίπτωση που η ερώτηση είναι της μορφής «τι κάνει ο/η/το Χ». Στην περίπτωση αυτή θα αναζητηθεί το ρήμα «κάνω» όπως όλα τα ρήματα. Αν δε βρεθεί θα αναζητηθεί σε γεγονός του RelModel/8 το entity ή entity type Χ και στη συνέχεια το ρήμα που είναι συνδεδεμένο με το Χ, στο οποίο (ρήμα) το Χ είναι υποκείμενο. Η απάντηση θα είναι της μορφής «Ο/Η/Το Χ Ρήμα ...». Αν για παράδειγμα έχει εισαχθεί το κείμενο «Η δασκάλα διδάσκει ιστορία» και στη συνέχεια η ερώτηση «Τι κάνει η δασκάλα;» θα εντοπιστεί ότι η «η δασκάλα διδάσκει» (γιατί το «δασκάλα» είναι υποκείμενο του «διδάσκει» στην πρόταση) οπότε για την εφαρμογή η ερώτηση μετασχηματίζεται σε «Τι διδάσκει η δασκάλα;» και η απάντηση θα είναι «Η δασκάλα διδάσκει ιστορία».

Για κάθε ερώτημα το σύστημα σταματάει την αναζήτηση μόλις εντοπίσει την πρώτη έγκυρη απάντηση. Για παράδειγμα για το κείμενο «Ο Γιώργος διαβάζει ένα βιβλίο. Η Μαρία διαβάζει ένα βιβλίο», στην ερώτηση «Ποιος διαβάζει βιβλίο;» θα δοθεί η απάντηση «Ο Γιώργος διαβάζει βιβλίο.» επειδή η αναζήτηση σταματάει με τον εντοπισμό της πρώτης οντότητας που ικανοποιεί το ερώτημα.

5.4.3 Δομικά κατηγορήματα

Τα κατηγορήματα με τα οποία υλοποιούνται οι βασικές λειτουργίες του αλγορίθμου δημιουργίας απαντήσεων είναι το **getAnswer/6** και το **getNounTree/3**. Το **getAnswer/6** παράγει προτάσεις σε φυσική γλώσσα σύμφωνα με τις παραμέτρους με τις οποίες καλείται ενώ το **getNounTree/3** δέχεται ως ορίσματα τα ορίσματα του γεγονότος του RelModel/8 το οποίο περιέχει ένα relation καθώς και το είδος της ερώτησης και εντοπίζει ανάλογα το υποκείμενο ή το αντικείμενο του ρήματος (relation) δημιουργώντας στη συνέχεια την απάντηση σε φυσική γλώσσα.

Το κατηγορήμα **getAnswer/6** δέχεται το ID του relation (ρήμα) και τη λίστα των IDs των συνδεδεμένων σε αυτό entity types/entities (ουσιαστικά, κύρια ονόματα) και δημιουργεί ένα κείμενο που περιέχει τα entities αυτά η μορφή του οποίου καθορίζεται από τις παραμέτρους κλήσης του κατηγορήματος. Το κείμενο αυτό χρησιμοποιείται ως υποκείμενο ή αντικείμενο της πρότασης της απάντησης. Το κατηγορήμα έχει τη μορφή

getAnswer(ID, Λίστα_Συνδεμένων_IDS, Πτώση, Εμφαν_Άρθρο, Εμφαν_Ουσιαστικό, Κείμενο).

Όπου

- «ID» είναι το ID του ρήματος του γεγονότος του RelModel/8 με Mtype=e (δομές προτάσεων).
- «Λίστα_Συνδεμένων» είναι μία λίστα που περιέχει τα Ids των γεγονότων του RelModel/8 που είναι συνδεδεμένα με το ID. Από τα IDs χρησιμοποιείται μόνο το πρώτο στη σύνθεση της απάντησης.
- «Πτώση» είναι η πτώση στην οποία θα δημιουργηθεί η απάντηση.
- Το «Εμφαν_Άρθρο» παίρνει τις τιμές inclArticle και exclArticle που δηλώνουν αν στη φράση που θα δημιουργηθεί θα συμπεριλαμβάνεται ή δεν θα συμπεριλαμβάνεται το άρθρο.
- Το «Εμφαν_Ουσιαστικό» παίρνει τις τιμές inclNoun και exclNoun που δηλώνουν αν στη φράση που θα δημιουργηθεί θα συμπεριλαμβάνεται ή δεν θα συμπεριλαμβάνεται το ουσιαστικό.
- Το «Κείμενο» είναι η φράση που δημιουργείται με βάση τις προηγούμενες παραμέτρους.

Παράδειγμα:

Στον Πίνακα 12 της ενότητας 5.3 φαίνονται οι τιμές των ορισμάτων των γεγονότων του RelModel/8 για την πρόταση «*Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά*»

Οι τιμές των 5 πρώτων ορισμάτων των γεγονότων του RelModel/8 επιδεικνύονται με μορφή πίνακα για καλύτερη μελέτη στον Πίνακα 16.

Id	RefTable	Ref Id	(Ρίζα)	ConnectedTo	Connect edType	LexType	WordGrammar	MTy pe
1	relationship	4	ειν	0	0	verb	[verb,ειν,trans,enikos,trito,10]	e
2	entity	1	υπογραφ	1	0	noun	[noun,υπογραφ,thiliko,enikos,onomastiki,100]	e
3	attribute	1	ψηφιακ	2	0	adjective	[adjective,ψηφιακ,thiliko,enikos,onomastiki,1]	e
4	entity	2	συστημ	1	1	noun	[noun,συστημ,oydetero,enikos,aitiatiki,203]	e
5	attribute	2	μαθηματικ	4	1	adjective	[adjective,μαθηματικ,oydetero,enikos,aitiatiki,1]	e

Πίνακας 16. Οι τιμές των 5 πρώτων ορισμάτων των γεγονότων του RelModel/8 με τη σημασιολογική αναπαράσταση της ερώτησης «*Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά*»

Η κλήση του κατηγορήματος **getAnswer/6** με τις ακόλουθες παραμέτρους

```
getAnswer(1, [2,4], onomastiki, inclArticle, inclNoun, Ans).
```

θα επιστρέψει

```
Ans = 'η ψηφιακή υπογραφή'.
```

Αν στο δεύτερο όρισμα η λίστα αλλάξει σε [4] (χρησιμοποιείται μόνο το πρώτο στοιχείο της λίστας) τότε η κλήση του getAnswer/6 με τις ακόλουθες παραμέτρους

```
getAnswer(1, [4], onomastiki, inclArticle, inclNoun, Ans).
```

Θα επιστρέψει

```
Ans = 'το μαθηματικό σύστημα'.
```

Κλήση του `getAnswer/6` με «Πτώση» `geniki`

```
getAnswer(1, [2,4], geniki, inclArticle, inclNoun, Ans).
```

```
Ans = 'της ψηφιακής υπογραφής'.
```

Κλήση του `getAnswer/6` με «Πτώση» `aitiatiki`

```
getAnswer(1, [2,4], aitiatiki, inclArticle, inclNoun, Ans).
```

```
Ans = 'την ψηφιακή υπογραφή'.
```

Κλήση του `getAnswer/6` με «Εμφαν_Άρθρο» `exclArticle` (αντί `inclArticle`)

```
getAnswer(1, [2,4], onomastiki, exclArticle, inclNoun, Ans).
```

```
Ans = 'ψηφιακή υπογραφή'.
```

Κλήση του `getAnswer/6` με «Εμφαν_Ουσιαστικό» `exclNoun` (αντί `inclNoun`)

```
getAnswer(1, [2,4], onomastiki, inclArticle, exclNoun, Ans).
```

```
Ans = 'η ψηφιακή'.
```

Με τους συνδυασμούς όλων των παραμέτρων μπορεί να δημιουργηθεί ο τύπος της απάντησης που ταιριάζει καλύτερα από άποψη γραμματικής και σύνταξης σε κάθε ερώτηση.

Το κατηγορήμα **`getNounTree/3`** δέχεται ως όρισμα το ID ενός γεγονότος του κατηγορήματος `relModel/8` με δείκτη σε ένα ρήμα και δημιουργεί μια φράση βασισμένη στο `entity type` (ουσιαστικό) το οποίο είναι συνδεδεμένο στο ρήμα ως υποκείμενο ή ως αντικείμενο. Η φράση περιέχει και τα συνδεδεμένα `attributes` (επίθετα και ουσιαστικά) στο `entity type`. Η επιλογή της εμφάνισης στο κείμενο του υποκειμένου ή του αντικειμένου καθορίζεται από το είδος της ερώτησης. Συγκεκριμένα η κλήση στο κατηγορήμα **`getNounTree/3`** συντάσσεται ως εξής:

`getNounTree(ID, Είδος_Ερώτησης, Κείμενο)`.

Όπου

- «ID» είναι το ID του ρήματος ενός γεγονότος του κατηγορήματος `relModel/8` με `Mtype=e` (δομές προτάσεων).
- Το «Είδος_Ερώτησης» παίρνει τις τιμές “ροίος” και “ti” ανάλογα με το είδος της ερώτησης.

- «Κείμενο» είναι η φράση που επιστρέφεται.

Παράδειγμα:

Οι τιμές των ορισμάτων 5 γεγονότων του κατηγορήματος relModel/8 για την πρόταση «*Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά*» φαίνονται σε μορφή πίνακα στον προηγούμενο πίνακα. Στο γεγονός τα ορίσματα του οποίου παριστάνονται στη γραμμή με ID=1 είναι καταχωρημένος ο δείκτης που δείχνει στο ρήμα «είναι» στο οποίο είναι συνδεδεμένο το «υπογραφή» ως υποκείμενο και το «σύστημα» ως αντικείμενο.

Η κλήση του **getNounTree/3** με ID=1 και Είδος_Ερώτησης=poios

```
getNounTree(1, poios, Ans).
```

επιστρέφει

```
Ans = 'η ψηφιακή υπογραφή'.
```

Η κλήση του **getNounTree/3** με ID=1 και Είδος_Ερώτησης=ti

```
getNounTree(1, ti, Ans).
```

επιστρέφει

```
Ans = 'μαθηματικό σύστημα'.
```


5.4.4 Παραδείγματα

Παράδειγμα 1

Στο παρακάτω παράδειγμα εισάγεται η πρόταση:

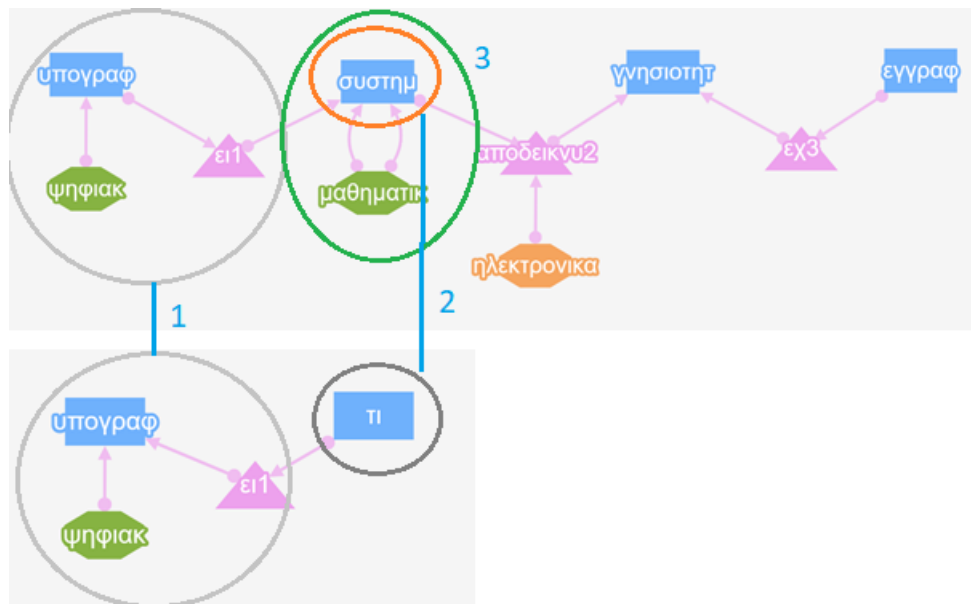
Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά

Στη συνέχεια μπορούν να απαντηθούν οι ερωτήσεις:

Ερώτηση 1:

Τι είναι η ψηφιακή υπογραφή;

Στο Σχήμα 27 φαίνεται γραφικά η αναπαράσταση των σημασιολογικών δομών του κειμένου της πρότασης και της ερώτησης που δημιουργήθηκε από τα γεγονότα του κατηγορήματος relModel/8 .



Σχήμα 27. Τα θήματα για την αναζήτηση της απάντησης στο ερώτημα «Τι είναι η ψηφιακή υπογραφή;»

Για να καθορισθεί η απάντηση ακολουθούνται τα βήματα (Σχήμα 27):

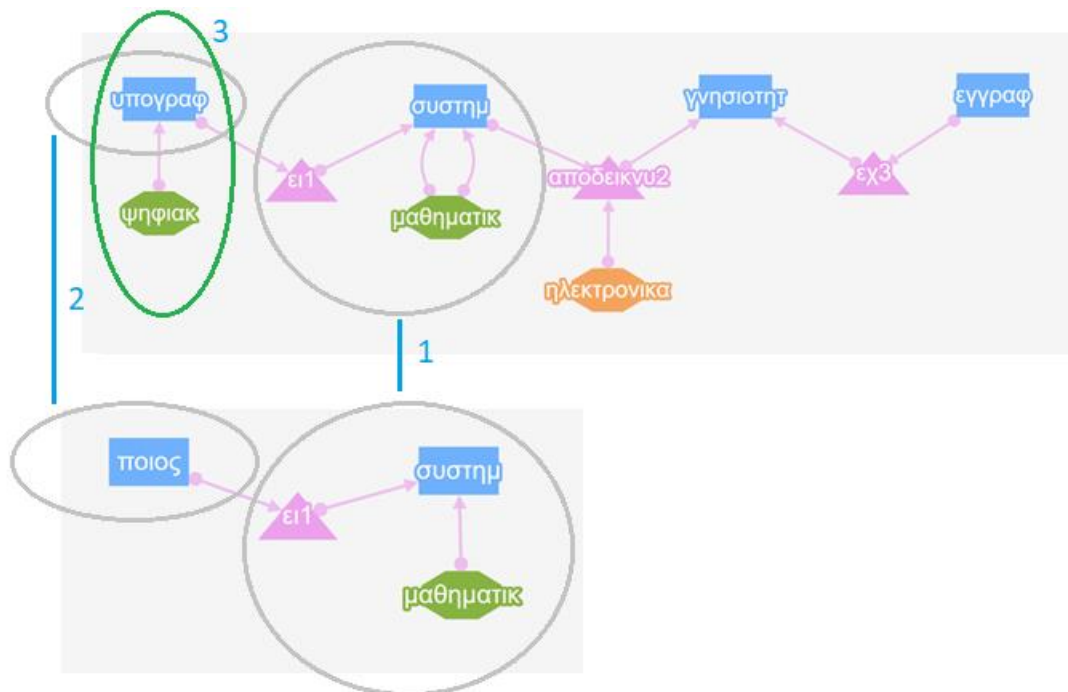
1. Συγκρίνεται η δομή της ερώτησης χωρίς το «τι» με αυτό της πρότασης. Εντοπίζεται μια δομή στον πίνακα σημασιολογικής δομής ίδια με αυτήν της ερώτησης.
2. Το «τι» καθορίζει ότι αναζητείται το αντικείμενο της πρότασης. Αναζητούμε στον πίνακα σημασιολογικής δομής το entity (ή entity type) που είναι συνδεδεμένο στο ρήμα «είναι» ως αντικείμενο. Η λέξη που εντοπίζεται είναι «σύστημα».
3. Εντοπίζονται τα attributes του entity type που είναι το «ψηφιακή» και δημιουργείται η απάντηση.

Η απάντηση είναι «**Η ψηφιακή υπογραφή είναι μαθηματικό σύστημα**»

Ερώτηση 2:

Ποιος είναι μαθηματικό σύστημα;

Στο Σχήμα 28 φαίνεται γραφικά η αναπαράσταση των σημασιολογικών δομών του κειμένου της πρότασης και της ερώτησης που δημιουργήθηκε από τα γεγονότα του κατηγορήματος relModel/8.



Σχήμα 28. Τα βήματα για την αναζήτηση της απάντησης στο ερώτημα «Ποιος είναι μαθηματικό σύστημα»

Για να καθορισθεί η απάντηση ακολουθούνται τα βήματα (Σχήμα 28):

1. Συγκρίνεται η δομή της ερώτησης χωρίς το «ποιος» με αυτό της πρότασης. Εντοπίζεται στα γεγονότα του relModel/8 μια σημασιολογική δομή ίδια με αυτήν της ερώτησης.
2. Το «**ποιος**» καθορίζει ότι αναζητείται το *υποκείμενο της πρότασης*. Εντοπίζεται στη δομή της βάσης το entity (ή entity type) που είναι συνδεδεμένο στο ρήμα «είναι» ως υποκείμενο. Η λέξη που εντοπίζεται είναι η «υπογραφή».
3. Εντοπίζονται τα attributes του entity type, που είναι το «ψηφιακή» και δημιουργείται η απάντηση.

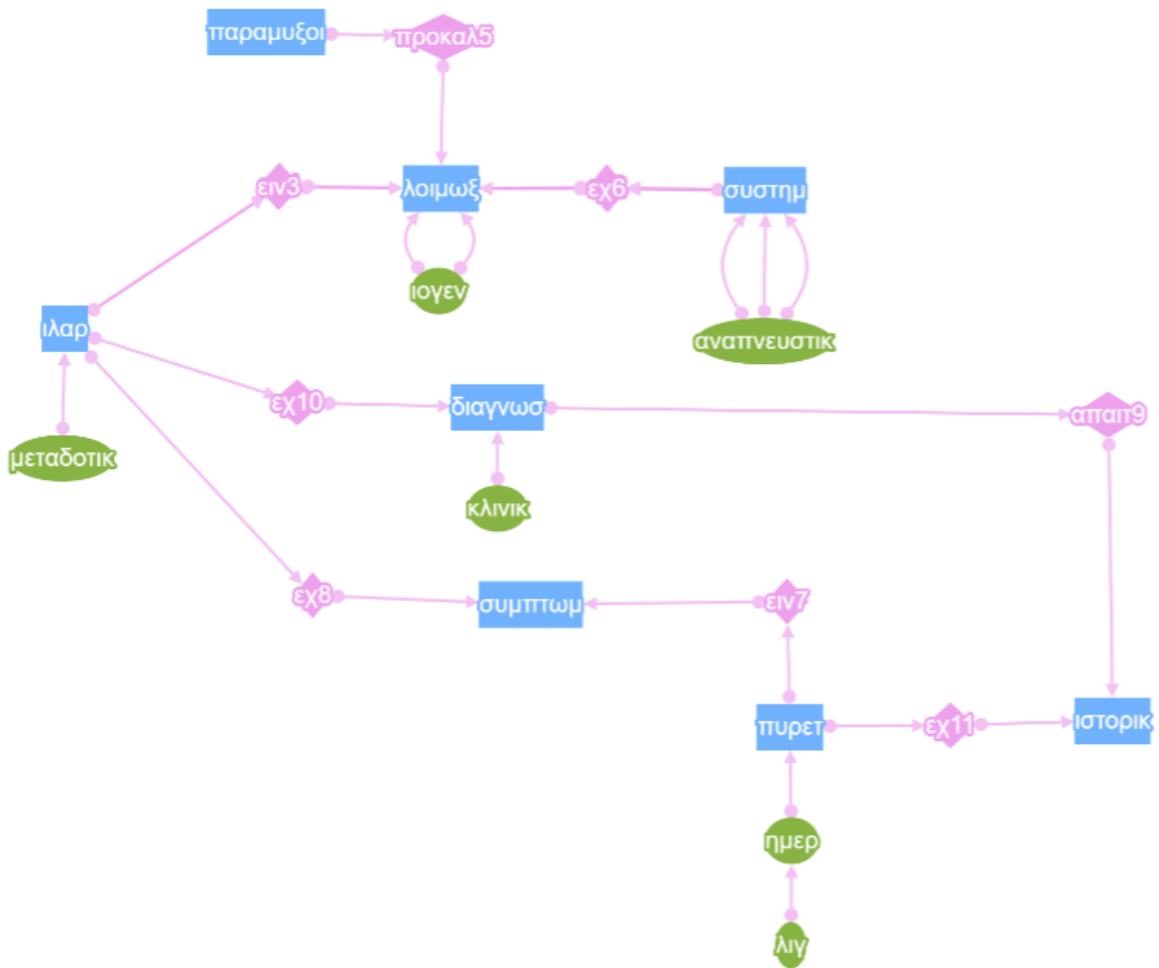
Απάντηση: «**Η ψηφιακή υπογραφή είναι μαθηματικό σύστημα**»

Παράδειγμα 2

Εισάγεται στη εφαρμογή το κείμενο:

*Η ιλαρά είναι μια ιογενής λοίμωξη του αναπνευστικού συστήματος.
Η ιλαρά είναι μεταδοτική.
Ο παραμυξιοδής προκαλεί τη λοίμωξη του αναπνευστικού συστήματος.
Ο πυρετός είναι σύμπτωμα της ιλαράς.
Το γενετικό υλικό του παραμυξιοδούς είναι το μονόκλωνο RNA.
Η κλινική διάγνωση της ιλαράς απαιτεί ιστορικό πυρετού λίγων ημερών.*

Στο σχεδιάγραμμα φαίνεται γραφικά η αναπαράσταση της σημασιολογικής δομής του κειμένου η οποία δημιουργήθηκε από τα γεγονότα του κατηγορήματος relModel/8 που αντιστοιχούν στο κείμενο του παραδείγματος.



Σχήμα 29. Γραφική αναπαράσταση της σημασιολογικής δομής του κειμένου «Ο μαθητής διαβάζει ένα βιβλίο. Η όμορφη Μαρία φοράει ένα κόκκινο κοντό φόρεμα. Ο Γιάννης τρέχει γρήγορα.»

Χρησιμοποιώντας τη δομή που δημιουργήθηκε μπορούν να απαντηθούν όλες οι σχετικές ερωτήσεις της μορφής «ποιος» και «τι». Για κάθε ερώτηση που εισάγεται δημιουργείται το αντίστοιχο κείμενο απάντησης με το οποίο αποκρίνεται το σύστημα, όπως φαίνεται παρακάτω.

ΕΡΩΤΗΣΕΙΣ**ΑΠΑΝΤΗΣΕΙΣ****Τι είναι η ιλαρά;***Η ιλαρά είναι ιογενής λοίμωξη του συστήματος***Ποιο είναι το σύμπτωμα;***Ο πυρετός είναι σύμπτωμα***Ποιος προκαλεί λοίμωξη του αναπνευστικού συστήματος;***Ο παραμυξιοίος προκαλεί λοίμωξη του αναπνευστικού συστήματος***Ποιος είναι ιογενής λοίμωξη;***Η ιλαρά είναι ιογενής λοίμωξη***Ποιος είναι ιογενής λοίμωξη αναπνευστικού συστήματος;***Η ιλαρά είναι ιογενής λοίμωξη αναπνευστικού συστήματος***Τι κάνει ο παραμυξιοίος;***Ο παραμυξιοίος προκαλεί τη λοίμωξη του συστήματος***Τι προκαλεί ο παραμυξιοίος;***Ο παραμυξιοίος προκαλεί τη λοίμωξη του συστήματος***Ποιο είναι το μονόκλωνο RNA;***Το γενετικό υλικό του παραμυξιοίου είναι το μονόκλωνο RNA***Τι απαιτεί η κλινική διάγνωση;***Η κλινική διάγνωση απαιτεί το ιστορικό του πυρετού***Ποιος έχει γενετικό υλικό;***Ο παραμυξιοίος έχει γενετικό υλικό***Ποιος απαιτεί ιστορικό πυρετού;***Η κλινική διάγνωση της ιλαράς απαιτεί ιστορικό πυρετού***Ποιος απαιτεί ιστορικό πυρετού ημερών;***Η κλινική διάγνωση της ιλαράς απαιτεί ιστορικό πυρετού ημερών***Ποιος απαιτεί ιστορικό πυρετού λίγων ημερών;***Η κλινική διάγνωση της ιλαράς απαιτεί ιστορικό πυρετού λίγων ημερών*

Η εφαρμογή μπορεί να επεκταθεί εύκολα ώστε να απαντάει σε ερωτήσεις τύπου «Ποιον», «Ποιου», «Πόσα» κλπ. Επίσης μπορεί να ζητάει διευκρινήσεις στις ερωτήσεις για τον εντοπισμό της ακριβούς απάντησης.

5.5 User Interface

5.5.1 Περιγραφή



Σχήμα 30. Η τοπολογία του συστήματος ερωταποκρίσεων Knowledge

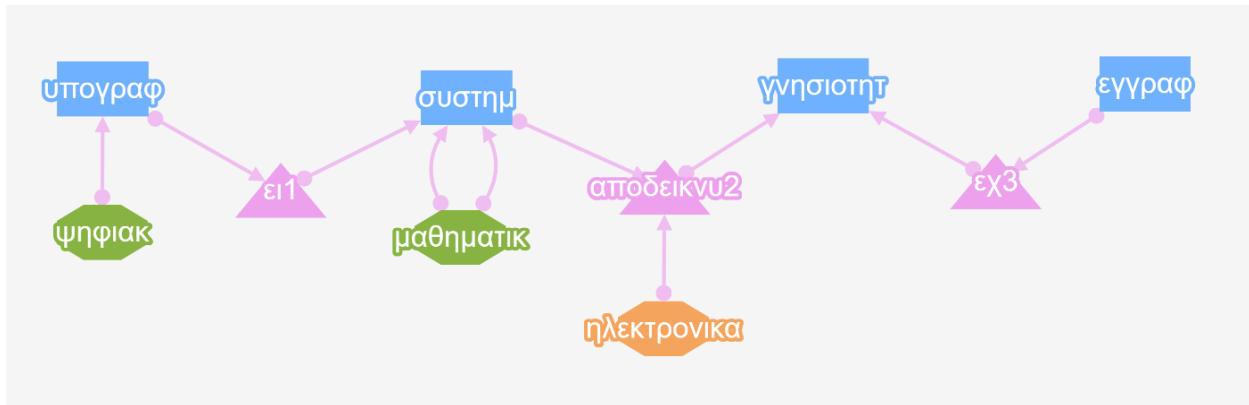
Η επικοινωνία του χρήστη με την εφαρμογή πραγματοποιείται μέσα από το web interface. Ο χρήστης έχει τη δυνατότητα να εισάγει κείμενο στον web browser το οποίο αναλύεται και η σημασιολογική δομή που δημιουργείται αποθηκεύεται στη βάση γνώση του συστήματος, ώστε να χρησιμοποιηθεί για επεξεργασία. Από αυτή την αναπαράσταση δημιουργείται η γραφική απεικόνιση των σημασιολογικών δομών των κειμένων, επιπλέον μπορεί να ζητηθούν πληροφορίες που περιέχονται στο λεξικό της εφαρμογής καθώς και να αναζητηθούν απαντήσεις σε ερωτήματα που θέτει ο χρήστης του συστήματος. Για την ανάπτυξη του web interface έχει αξιοποιηθεί η δυνατότητα βιβλιοθηκών της SWI Prolog για υλοποίηση HTTP server καθώς και η δυνατότητα για προσπέλαση δεδομένων στον HTTP server. Η γραφική απεικόνιση στον browser γίνεται από με τη χρήση της βιβλιοθήκης “Cytoscape”, η οποία είναι γραμμένη σε κώδικα javascript και παρέχει μεταξύ άλλων συναρτήσεις απεικόνισης κόμβων και συνδέσεων. Επιπλέον χρησιμοποιήθηκε το Bootstrap που είναι μια συλλογή από εργαλεία HTML, CSS, και JS για τη δημιουργία ιστοσελίδων οι οποίες εμφανίζονται σωστά σε διαφορετικές αναλύσεις οθόνης.

5.5.2 Γραφική αναπαράσταση της δομής των προτάσεων

Η σημασιολογική δομή των προτάσεων που είναι αποθηκευμένες στη βάση γνώσης αναπαριστάται γραφικά στον browser. Για την απεικόνιση των δεδομένων στον browser χρησιμοποιείται κώδικας javascript με τον οποίο ζητούνται τα δεδομένα από τον Web Server. Ο κώδικας της Prolog στον server δημιουργεί από τα γεγονότα του κατηγορήματος `relModel/8` κώδικα Javascript στον οποίο η γνώση που υπάρχει στα γεγονότα του `relModel/8` αναπαρίστανται με κόμβους και οι συνδέσεις τους με διανύσματα. Για τη δημιουργία του διαγράμματος που εμφανίζεται στο χρήστη εφαρμόζονται ορισμένοι κανόνες, όπως π.χ. παρουσίαση όλων των *entities* και *attributes* με την ίδια ρίζα λέξης ως ένα αντικείμενο και όχι ξεχωριστά, ώστε η γραφική αναπαράσταση να είναι πιο συνοπτική και κατανοητή. Τα **entities** απεικονίζονται με μπλε χρώμα σε τετράγωνο, τα **entity types** με γκριζο χρώμα σε τετράγωνο, τα **attributes of entities** με πράσινο χρώμα σε εξάγωνο, τα **relationships** με μωβ χρώμα σε τρίγωνο και τα **attributes of relationships** με πορτοκαλί χρώμα σε εξάγωνο.

Παράδειγμα: Στο Σχήμα 31 φαίνεται γραφικά η αναπαράσταση που δημιουργήθηκε από τα γεγονότα του κατηγορήματος relModel/8 για τη πρόταση:

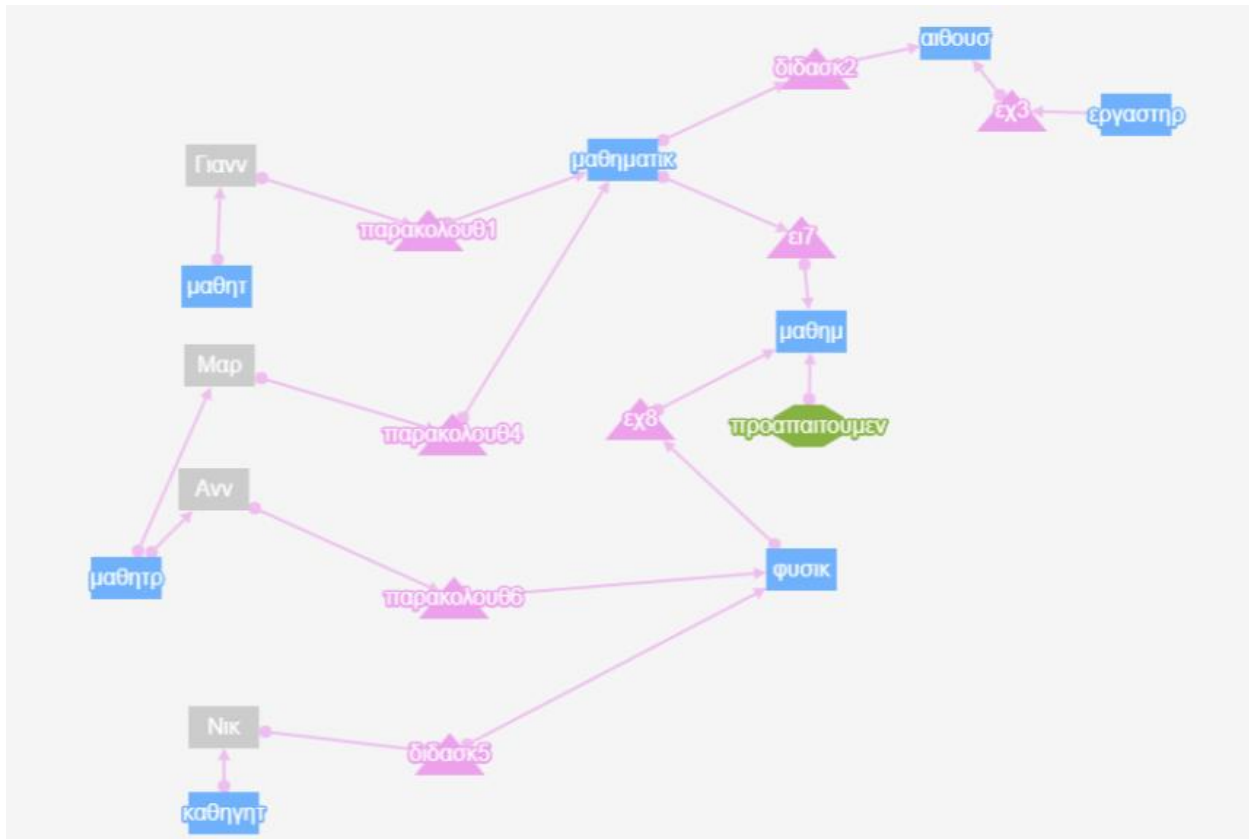
Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά



Σχήμα 31. Η αναπαράσταση της σημασιολογικής δομής της πρότασης «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά» όπως εμφανίζεται στον χρήστη

Παράδειγμα: Στο Σχήμα 32 φαίνεται γραφικά η αναπαράσταση που δημιουργήθηκε από τα γεγονότα του κατηγορήματος relModel/8 για τη πρόταση.

*Ο καθηγητής Γιώργος διδάσκει μαθηματικά.
Ο μαθητής Γιάννης παρακολουθεί μαθηματικά.
Τα μαθηματικά διδάσκονται στην αίθουσα του εργαστήριου.
Η μαθήτρια Μαρία παρακολουθεί μαθηματικά.
Ο καθηγητής Νίκος διδάσκει φυσική.
Η μαθήτρια Άννα παρακολουθεί φυσική.
Τα μαθηματικά είναι προαπαιτούμενο μάθημα της φυσικής*



Σχήμα 32. Η αναπαράσταση της σημασιολογικής δομής της πρότασης «Ο καθηγητής Γιώργος διδάσκει μαθηματικά. Ο μαθητής Γιάννης παρακολουθεί μαθηματικά...» όπως εμφανίζεται στο χρήστη

5.5.3 SWI Prolog Web Server

Η δημιουργία ενός Web Server ο οποίος αποκρίνεται σε συγκεκριμένο port στην SWI Prolog απαιτεί ελάχιστες γραμμές κώδικα. Συγκεκριμένα πρέπει να συμπεριληφθούν στον κώδικα τα modules τα οποία απαιτούνται για τον Server

```
:- use_module(library(http/thread_httpd)).
:- use_module(library(http/http_dispatch)).
```

Το κατηγορημα

```
server(Port) :- http_server(http_dispatch, [port(Port)]).
```

εκκινεί τον Web Server στο δεδομένο Port. Έχοντας συμπεριλάβει τις προηγούμενες γραμμές κώδικα στην εφαρμογή, το κατηγορημα

```
server(8000).
```


εκκινεί τον Web Server ο οποίος κάνει "listen" στο port 8000. Ο χρήστης είναι σε θέση να επικοινωνήσει με τον Web Server ανοίγοντας έναν web browser και πληκτρολογώντας στη διεύθυνση το url "http://127.0.0.1:8000/" ή όμοια "http://localhost:8000/". Τα αιτήματα του χρήστη αποστέλλονται στον server όπου επεξεργάζονται και δημιουργείται η αντίστοιχη σελίδα html η οποία εμφανίζεται στον browser.

5.5.4 HTTP Request handling

Το κείμενο που εισάγεται στον browser από το χρήστη, αποστέλλεται στο Web Server μέσα από ένα Post Request. Για να διαχειριστεί το request ο Web Server έχει συμπεριληφθεί στον κώδικα της εφαρμογής η γραμμή

```
:- http_handler('/main', main_pad, []).
```

η οποία δηλώνει ότι όταν ζητηθεί από το χρήστη η σελίδα "/main", δηλαδή όταν το url στον browser είναι το "http://localhost:8000/main" το request θα καλέσει το κατηγορήμα **main_pad**. Μέσα στο κατηγορήμα θα ληφθούν τα δεδομένα "Data" του Post μέσα από τις δύο πρώτες γραμμές του κώδικα του κατηγορήματος όπως φαίνεται παρακάτω και ακολούθως το κείμενο που εισήγαγε ο χρήστης (Data.inputText) θα σταλεί μέσα από την εντολή

```
processInput(Data.inputText, responseText)
```

για επεξεργασία από την εφαρμογή. Μετά την επεξεργασία θα επιστρέψει το responseText το οποίο είναι το κείμενο που θα εμφανιστεί ως απάντηση στο χρήστη. Στη συνέχεια θα δημιουργηθεί με την reply_html_page η σελίδα HTML που θα εμφανιστεί στο χρήστη στην οποία περιλαμβάνονται μεταξύ των άλλων ο τίτλος της σελίδας (<title>), οι φόρμες που περιέχει (<form>) και τα scripts που φορτώνονται (<script>) για να λειτουργήσει σωστά η σελίδα.

```
main_pad(Request) :-  
    member(method(post), Request),  
    http_read_data(Request, Data, []),  
    processInput(Data.inputText, S),  
    reply_html_page(  
        ..
```

5.5.5 Ενσωμάτωση και εκτέλεση κώδικα στη σελίδα HTML

Στη σελίδα HTML η οποία εμφανίζεται στον browser του χρήστη περιέχεται ο κώδικας javascript και css από τις βιβλιοθήκες ανοιχτού κώδικα Bootstrap και Cytoscape που χρησιμοποιούνται για τη σωστή εμφάνιση και τη λειτουργία της σελίδας και τη γραφική απεικόνιση των δομών που είναι αποθηκευμένες στη βάση. Για να φορτωθεί ο κώδικας css χρησιμοποιείται μέσα στην reply_html_page της προηγούμενης παραγράφου η εντολή

```
html_requires (PathToFileList)
```

όπου `PathToFileList` είναι μία λίστα με τα `uri` των αρχείων με κατάληξη `.css` που θα φορτωθούν μαζί με τη σελίδα.

Όμοια για να φορτωθεί ο κώδικας javascript χρησιμοποιείται η εντολή

```
script(src(PathToFile), [])
```

όπου `PathToFile` είναι το `uri` του κάθε αρχείου με κατάληξη `.js` που θα φορτωθεί μαζί με τη σελίδα όπως φαίνεται στον κώδικα που ακολουθεί.

```
reply_html_page(  
  title('Entities recognizer'), %Page title  
  [  
    \html_requires([css('bootstrap.min.css'), css('core.css')]),  
    ..  
    script(src("/js/bootstrap.min.js"), []),  
    script(src("/js/cytoscape.min.js"), []),  
    script(src("/js/scripts.js"), [])  
  ]  
)
```

Το αρχείο `scripts.js` περιέχει τον κώδικα javascript που έχει αναπτυχθεί για την αναπαράσταση της δομής της βάσης. Ο κώδικας τρέχει μόλις ολοκληρωθεί το φόρτωμα της σελίδας HTML και των στοιχείων που περιέχονται σε αυτήν. Μέσα στη σελίδα HTML μετά από κάθε Post Request περιέχονται αποθηκευμένα σε μεταβλητές της javascript τα δεδομένα για κάθε στοιχείο και κάθε σύνδεση που υπάρχουν καταχωρημένα στη βάση γνώσης στη μορφή:

```
<script>var items =[{name:N1, id:I1, type:T1},{name:N2, id:I2,  
type:T2},...]</script>
```

```
<script>var connections =[{from:Isrt_1, to:Iend_1, type:CT1}, {from:Isrt_2,  
to:Iend_2, type:CT2},... ]</script>
```

Στη μεταβλητή **items** το `Nx` είναι το όνομα του κόμβου `x`, `Ix` είναι ο αριθμός με τον οποίο θα γίνεται αναφορά στο συγκεκριμένο στοιχείο και `Tx` ο τύπος του (για `x=1,2,3..`). Το `id:Ix` χρησιμοποιείται μόνο για τα ρήματα διαφορετικά παραλείπεται. Το όνομα του κόμβου περιέχει τη ρίζα της λέξης (και έναν αύξοντα αριθμό εφόσον πρόκειται για ρήμα), ενώ ο τύπος καθορίζει το χρώμα και το σχήμα του. Οι τύποι των στοιχείων φαίνονται στον Πίνακα 17.

Όνομα τύπου	Αναπαριστά	Σχήμα	Χρώμα
ent0	Entity	Ορθογώνιο	Γκρι
ent	Entity Type	Ορθογώνιο	Μπλε
rel	Relationship	Ρόμβος	Μωβ
attr-ent	Attribute of an Entity	Έλλειψη	Πράσινο
attr-rel	Attribute of Relationship	Οκτάγωνο	Πορτοκαλί

Πίνακας 17. Τα χαρακτηριστικά της απεικόνισης των στοιχείων του Μοντέλου Οντοτήτων – Συσχετίσεων όπως εμφανίζονται στον χρήστη.

Στη μεταβλητή **connections** το `lsrt_x` είναι το "name" ή το "id" του κόμβου από τον οποίο ξεκινάει η σύνδεση `x` και `lend_x` είναι το "name" ή το "id" του κόμβου στον οποίο καταλήγει η σύνδεση `x` (για `x=1,2,3..`). Το `CTx` (connection type) καθορίζει την εμφάνιση της σύνδεσης, η οποία θα χρησιμοποιηθεί σε επόμενες εκδόσεις της εφαρμογής. Στην παρούσα έκδοση οι γραμμές που ενώνουν τα στοιχεία ξεκινούν από το στοιχείο που δηλώνεται στο *from* και καταλήγουν στο στοιχείο που δηλώνεται στο *to* και έχουν όλες χρώμα μωβ.

Με τη χρήση των δεδομένων από τις μεταβλητές **items** και **connections** καλούνται οι συναρτήσεις της βιβλιοθήκης `Cytospace` και σχεδιάζονται τα στοιχεία και οι συνδέσεις τους. Τα στοιχεία κατανομούνται αυτόματα σε όλη την έκταση του παραθύρου. Ο χρήστης έχει τη δυνατότητα να μετακινήσει τα στοιχεία με το ποντίκι διατηρώντας τις μεταξύ τους συνδέσεις ώστε να επιτύχει την καλύτερη δυνατή διάταξη. Επιπλέον έχει τη δυνατότητα να κάνει `zoom in` και `zoom out` ώστε να εστιάσει σε συγκεκριμένα σημεία ή να έχει συνολική εικόνα της δομής.

Παράδειγμα 1: Ο χρήστης πληκτρολογεί τη φράση «*Ο μαθητής διαβάζει ένα βιβλίο*» και αποστέλλει τη φόρμα με `POST Request` στον `Web Server`. Στον κώδικα `HTML` της σελίδας που φορτώνεται μετά το `POST` περιέχονται οι γραμμές:

```
<script>
    var items =[
        {name:"μαθητ", type:"ent"},
        {name:"βιβλ", type:"ent"},
        {name:"διαβαζ1", id:"1", type:"rel"}
    ]
</script>
<script>
    var connections =[
        {from:"μαθητ", to:"1", type:"entityrelationship"},
```

```
{from:"1", to:"βιβλ", type:"relationshipentity"}
```

```
]
```

```
</script>
```

Η μεταβλητή **items** περιέχει τα στοιχεία που εμφανίζονται στη γραφική αναπαράσταση της βάσης που είναι το «μαθητ», «βιβλ» και «διαβαζ1». Επειδή το «διαβαζ1» είναι ρήμα, το στοιχείο περιέχει id (id:"1") ενώ το όνομα του στοιχείου του είναι το θέμα του ρήματος ακολουθούμενο από το id («διαβαζ1»). Το «μαθητ» και «βιβλ» θα εμφανιστούν ως τύπου "ent" (entity) ενώ το «διαβαζ1» θα εμφανιστεί ως τύπου "rel" (relationship).

Η μεταβλητή **connections** περιέχει τις δύο συνδέσεις μεταξύ των στοιχείων. Η πρώτη σύνδεση ξεκινάει από το «μαθητ» και καταλήγει στο στοιχείο με id 1 δηλαδή το «διαβαζ1» (Σχήμα 33). Η δεύτερη ξεκινάει από το στοιχείο με id 1 και καταλήγει στο στοιχείο με όνομα «βιβλ».

Entities recognizer

Show/Hide Map

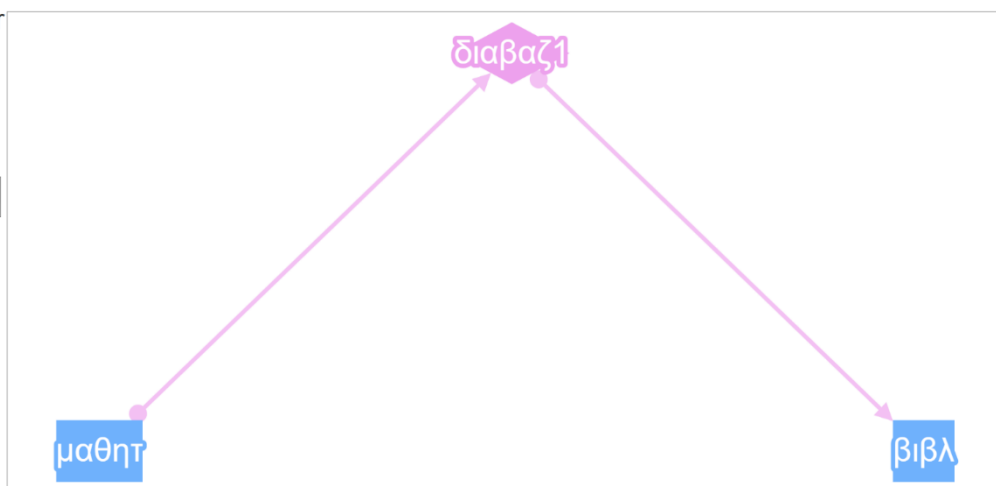
Φράση: ο μαθητής διαβάζει ένα βιβλ

Submit

Clear

Τι ξέρεις:

ο μαθητής διαβάζει ένα βιβλίο

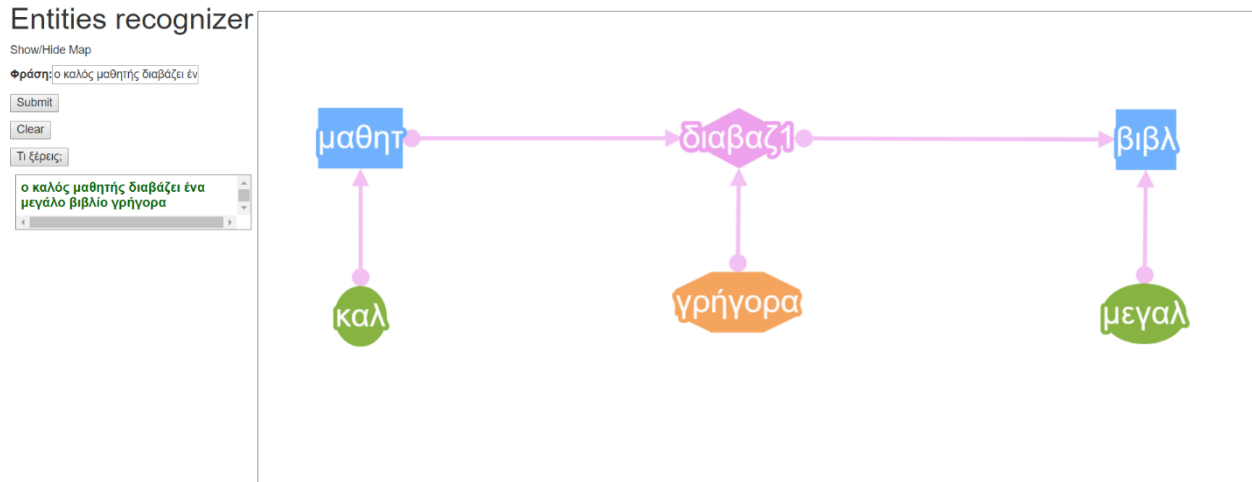


Σχήμα 33. Η έξοδος της εφαρμογής για το κείμενο «Ο μαθητής διαβάζει ένα βιβλίο»

Παράδειγμα 2: Ο χρήστης πληκτρολογεί τη φράση «Ο καλός μαθητής διαβάζει ένα μεγάλο βιβλίο γρήγορα» και αποστέλλει τη φόρμα με POST Request στον Web Server. Στον κώδικα HTML της σελίδας που φορτώνεται μετά το POST περιέχονται οι γραμμές:

```
<script>
    var items =[
        {name:"μαθητ", type:"ent"},
        {name:"βιβλ", type:"ent"},
        {name:"διαβαζ1", id:"1", type:"rel"},
        {name:"καλ", type:"attr-ent"},
        {name:"μεγαλ", type:"attr-ent"},
        {name:"γρήγορα", type:"attr-rel"}
    ]
</script>
<script>
    var connections =[
        {from:"μαθητ", to:"1", type:"entityrelationship"},
        {from:"καλ", to:"μαθητ", type:"attributeentity"},
        {from:"1", to:"βιβλ", type:"relationshipentity"},
        {from:"μεγαλ", to:"βιβλ", type:"attributeentity"},
        {from:"γρήγορα", to:"1", type:"relattrrelationship"}
    ]
</script>
```

Αρχικά τοποθετούνται τα 6 στοιχεία της δομής «μαθητ», «βιβλ», «διαβαζ1», «καλ», «μεγαλ», «γρηγορ» στο παράθυρο, καθένα από τα οποία εμφανίζεται σύμφωνα με τον τύπο του όπως αναφέρεται στη μεταβλητή **items**. Στη συνέχεια τα στοιχεία ενώνονται με 5 συνδέσεις με τη φορά που αναφέρεται στη μεταβλητή **connections**. Η γραφική απεικόνιση της δομής της βάσης που προκύπτει από τα δεδομένα φαίνεται στο *Σχήμα 34*.



Σχήμα 34. Η έξοδος της εφαρμογής για το κείμενο «Ο καλός μαθητής διαβάζει ένα μεγάλο βιβλίο γρήγορα»

5.5.6 Εντολές ελέγχου δεδομένων λεξικών

Η εφαρμογή μπορεί να δεχτεί συγκεκριμένες λέξεις ή φράσεις ως εντολές για τη διαχείρισή της ή για την ενημέρωση του χρήστη. Αν εισαχθεί η λέξη Clear στο πλαίσιο κειμένου, η εφαρμογή καθαρίζει όλες τα γεγονότα από τη βάση γνώσης. Αν εισαχθεί η φράση «Τι ξέρεις» η εφαρμογή επιστρέφει όλες τις λέξεις που περιέχονται στα λεξικά της όπως φαίνεται παρακάτω:

Ρήματα αμετάβατα: περπατώ, ανοίγω, κλειδώνω, τρέχω, παίζω, γράφω...

Ρήματα μεταβατικά: διαβάζω, δίνω, διδάσκω, παρακολουθώ, παρέχω, διαθέτω...

Ουσιαστικά αρσ: μαθητής, βαθμός, πυρετός, αριθμός, χρόνος, υπάλληλος,...

Ουσιαστικά θηλ: γυναίκα, δασκάλα, γραμμή, γνώση, διάγνωση, παραγωγή, απόφαση,...

Ουσιαστικά ουδ: βιβλίο, συμβούλιο, ελατήριο, εργαστήριο, επιτόκιο, εγχειρίδιο, καπέλο,...

Επίθετα: μεγάλος, μεγάλη, μεγάλο, κοντός, κοντή, κοντό, προαπαιτούμενος, προαπαιτούμενη, προαπαιτούμενο, κινητικός, κινητική, κινητικό, μικρός, μικρή, μικρό, κόκκινος, κόκκινη, κόκκινο, ...

Επιρρήματα: εδώ, εκεί, γρήγορα, αμέσως, τώρα, πολύ, δύσκολα, εξαιρετικά, ηλεκτρονικά, σήμερα, καθημερινά, αυτόματα, ...

5.6 Χρήση του user interface

Η χρήση του interface είναι σχετικά απλή. Εφόσον η εφαρμογή της Prolog τρέχει ήδη και κάνει "listen" στο port 8000, ο χρήστης ανοίγει έναν web browser και πληκτρολογεί στη διεύθυνση το url

http://127.0.0.1:8000

για να ανοίξει η σελίδα της εφαρμογής. Στη σελίδα εμφανίζεται ένα πλαίσιο κειμένου όπου ο χρήστης συμπληρώνει το κείμενο και πατάει Submit. Το κείμενο επεξεργάζεται από την εφαρμογή και στη συνέχεια εμφανίζεται η σελίδα που αναπαριστά τη δομή της πρότασης. Αν ο χρήστης εισάγει και δεύτερη πρόταση, η δομή της θα προστεθεί στην υπάρχουσα και θα εμφανιστεί η αναπαράσταση της

συνολικής δομής. Εναλλακτικά μπορεί να εισάγει πολλές προτάσεις για επεξεργασία, οπότε θα δημιουργηθεί η δομή που αντιστοιχεί στο σύνολο των προτάσεων.

Αν κάποια από τις λέξεις της πρότασης που εισάγεται δεν υπάρχει στο λεξικό της εφαρμογής ή περιέχει ορθογραφικά λάθη, η εφαρμογή θα επιστρέψει το μήνυμα «Άγνωστη λέξη:» μαζί με τη λέξη που δεν αναγνώρισε, ώστε ο χρήστης να τη διορθώσει ή να την προσθέσει στο λεξικό και να ξαναισάγει την πρόταση για επεξεργασία.

Στο παράθυρο της εφαρμογής το κουμπί “Clear” στέλνει την αντίστοιχη εντολή για διαγραφή όλων των γεγονότων της βάσης γνώσης. Ο χρήστης μπορεί επίσης να πληκτρολογήσει Clear στο πλαίσιο κειμένου και να έχει το ίδιο αποτέλεσμα. Όμοια ο χρήστης μπορεί να πατήσει το κουμπί «*Τι ξέρεις;*» ή να πληκτρολογήσει τη φράση «*Τι ξέρεις;*» στο πλαίσιο κειμένου και να πάρει την αναφορά με το περιεχόμενο των λεξικών της εφαρμογής.

Αν ο χρήστης εισάγει μια ερώτηση, η ερώτηση επεξεργάζεται και εμφανίζεται η απάντηση εφόσον υπάρχει η απαιτούμενη πληροφορία στη βάση γνώσης. Διαφορετικά εμφανίζεται το μήνυμα «*Λυπάμαι. Δεν γνωρίζω.*».

6. Παραδείγματα από τη χρήση του συστήματος

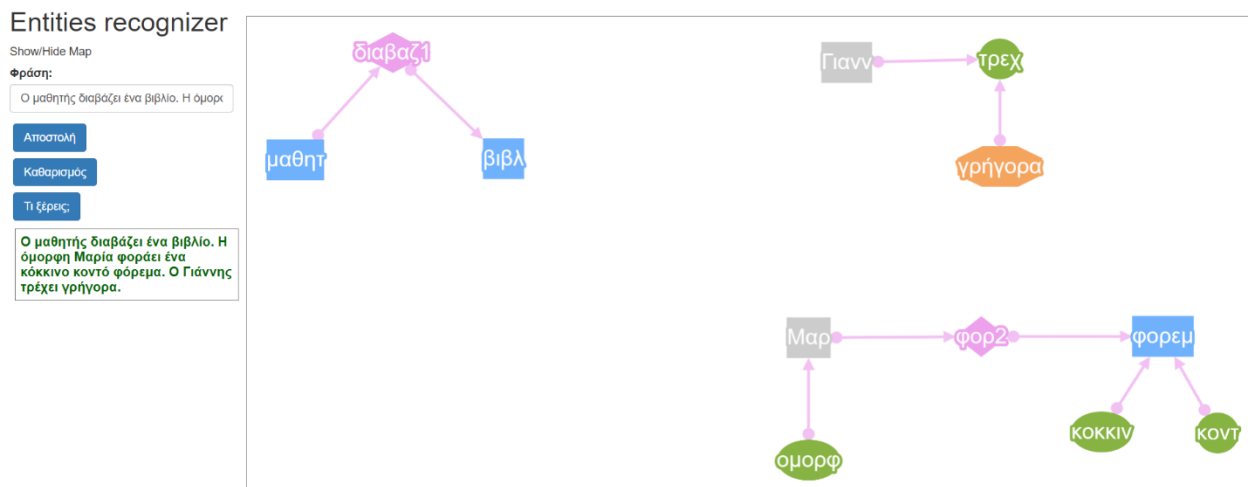
Η εφαρμογή παρέχει στο χρήστη τη δυνατότητα να εισάγει κείμενα και να βλέπει την αναπαράστασή τους, να θέτει ερωτήματα και να λαμβάνει την απόκριση του συστήματος και να ζητάει πληροφορίες για τα δεδομένα του λεξικού. Η σύνδεση στην εφαρμογή γίνεται μέσα από έναν web browser στο url <http://localhost:8000>.

Παράδειγμα 1

Στο πλαίσιο κειμένου με τον τίτλο «Φράση» ο χρήστης πληκτρολογεί το κείμενο «Ο μαθητής διαβάζει ένα βιβλίο. Η όμορφη Μαρία φοράει ένα κόκκινο κοντό φόρεμα. Ο Γιάννης τρέχει γρήγορα.» και πατάει το κουμπί «Αποστολή». Τα περιεχόμενα της σελίδας που θα φορτώσει φαίνονται στο Σχήμα 35. Αριστερά στην οθόνη φαίνεται η φράση και δεξιά της η γραφική αναπαράσταση της δομής του κειμένου.

Στη συνέχεια ο χρήστης εισάγει ερωτήσεις στο σύστημα σχετικά με το κείμενο. Στο Σχήμα 36 φαίνεται η έξοδος της εφαρμογής σε τέσσερα διαφορετικά ερωτήματα.

- Στο ερώτημα «Ποιος φοράει κοντό φόρεμα» το σύστημα αποκρίνεται «Η Μαρία φοράει κοντό φόρεμα».
- Στο ερώτημα «Ποιος τρέχει» το σύστημα αποκρίνεται «Ο Γιάννης τρέχει».
- Στο ερώτημα «Ποιος φοράει πράσινο φόρεμα» το σύστημα αποκρίνεται «Λυπάμαι. Δεν γνωρίζω», εφόσον δεν υπάρχει τέτοιου είδους πληροφορία καταχωρημένη στη βάση γνώση.
- Στο ερώτημα «Ποιος τρέχει καθημερινά» το σύστημα αποκρίνεται επίσης «Λυπάμαι. Δεν γνωρίζω», εφόσον αυτή η πληροφορία δεν είναι καταχωρημένη στη βάση γνώση.



Σχήμα 35. Τα περιεχόμενα της σελίδας της εφαρμογής για το κείμενο εισόδου: «Ο μαθητής διαβάζει ένα βιβλίο. Η όμορφη Μαρία φοράει ένα κόκκινο κοντό φόρεμα. Ο Γιάννης τρέχει γρήγορα»

Entities recognizer

Show/Hide Map

Φράση:

Ποιος φοράει κοντό φόρεμα;

Αποστολή

Καθαρισμός

Τι ξέρεις;

Ποιος φοράει κοντό φόρεμα;

η όμορφη Μαρία φοράει κοντό φόρεμα

Entities recognizer

Show/Hide Map

Φράση:

Ποιος τρέχει;

Αποστολή

Καθαρισμός

Τι ξέρεις;

Ποιος τρέχει;

ο Γιάννης τρέχει

Entities recognizer

Show/Hide Map

Φράση:

ποιος φοράει πράσινο φόρεμα;

Αποστολή

Καθαρισμός

Τι ξέρεις;

ποιος φοράει πράσινο φόρεμα;

Λυπάμαι. Δε γνωρίζω

Entities recognizer

Show/Hide Map

Φράση:

ποιος τρέχει καθημερινά;

Αποστολή

Καθαρισμός

Τι ξέρεις;

ποιος τρέχει καθημερινά;

Λυπάμαι. Δε γνωρίζω

Σχήμα 36. Οι απαντήσεις σε διαφορετικά ερωτήματα αφότου δοθεί το κείμενο εισόδου: «Ο μαθητής διαβάζει ένα βιβλίο. Η όμορφη Μαρία φοράει ένα κόκκινο κοντό φόρεμα. Ο Γιάννης τρέχει γρήγορα»

Παράδειγμα 2

Στο πλαίσιο κειμένου με τον τίτλο «Φράση» ο χρήστης πληκτρολογεί το κείμενο με ένα ορθογραφικό λάθος: «Ο μαθτής διαβάζει ένα βιβλίο» και πατάει το κουμπί «Αποστολή». Τα περιεχόμενα της σελίδας που θα φορτώσει φαίνονται στο Σχήμα 37. Η απόκριση του συστήματος είναι «Άγνωστη λέξη: μαθτής». Ο χρήστης θα πρέπει να διορθώσει το ορθογραφικό λάθος και να πατήσει ξανά το κουμπί «Αποστολή».

Entities recognizer

Show/Hide Map

Φράση:

Ο μαθτής διαβάζει ένα βιβλίο

Αποστολή

Καθαρισμός

Τι ξέρεις;

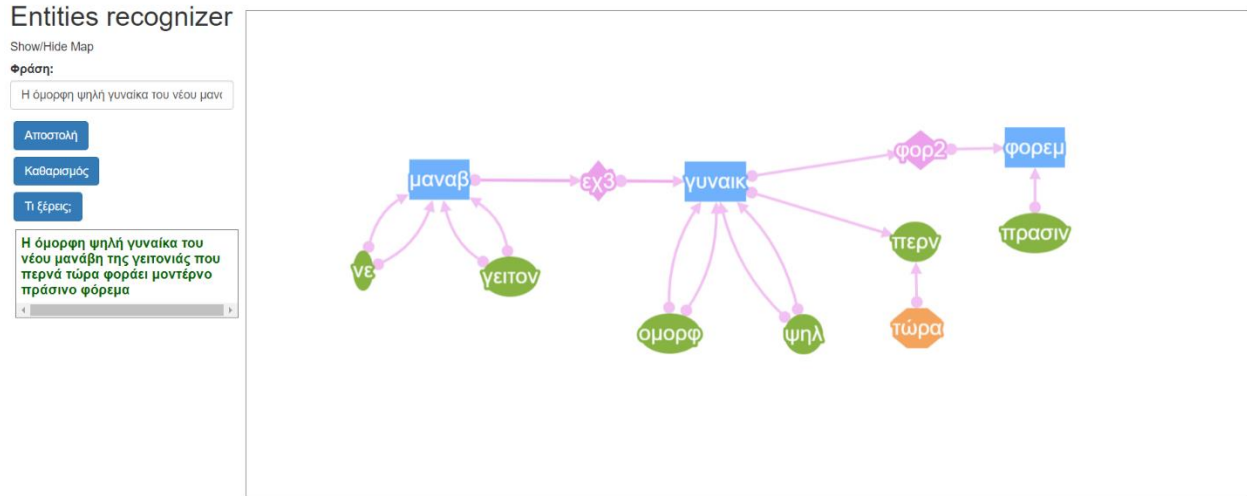
Ο μαθτής διαβάζει ένα βιβλίο

Άγνωστη λέξη: **μαθτής**

Σχήμα 37. Τα περιεχόμενα της σελίδας της εφαρμογής για το κείμενο εισόδου με ορθογραφικό λάθος: «Ο μαθτής διαβάζει ένα βιβλίο»

Παράδειγμα 3

Στο πλαίσιο κειμένου με τον τίτλο «Φράση» ο χρήστης πληκτρολογεί το κείμενο «*Η όμορφη ψηλή γυναίκα του νέου μανάβη της γειτονιάς που περνά τώρα φοράει μοντέρνο πράσινο φόρεμα*» και πατάει το κουμπί «Αποστολή». Τα περιεχόμενα της σελίδας που θα φορτώσει φαίνονται στο Σχήμα 38. Αριστερά στην οθόνη φαίνεται η φράση και δεξιά της η γραφική αναπαράσταση της δομής του κειμένου.



Σχήμα 38. Τα περιεχόμενα της σελίδας της εφαρμογής για το κείμενο εισόδου: «*Η όμορφη ψηλή γυναίκα του νέου μανάβη της γειτονιάς που περνά τώρα φοράει μοντέρνο πράσινο φόρεμα*»

Στη συνέχεια ο χρήστης εισάγει ερωτήσεις στο σύστημα σχετικά με το κείμενο. Στο Σχήμα 39 φαίνεται η έξοδος της εφαρμογής σε τέσσερα διαφορετικά ερωτήματα.

- Στο ερώτημα «*Τι κάνει η γυναίκα του μανάβη;*» το σύστημα αποκρίνεται «*Η γυναίκα του μανάβη περνά τώρα*».
- Στο ερώτημα «*Ποιος περνά;*» το σύστημα αποκρίνεται «*Η όμορφη ψηλή γυναίκα του μανάβη περνά*».
- Στο ερώτημα «*Τι έχει ο μανάβης;*» το σύστημα αποκρίνεται «*Ο μανάβης έχει όμορφη ψηλή γυναίκα*».
- Στο ερώτημα «*Τι φοράει ο μανάβης;*» το σύστημα αποκρίνεται «*Λυπάμαι. Δε γνωρίζω*» εφόσον δεν έχει αποθηκευμένη τέτοια πληροφορία.

Entities recognizer

Show/Hide Map

Φράση:

Αποστολή

Καθαρισμός

Τι ξέρεις;

η γυναίκα του μανάβη περνά τώρα

Entities recognizer

Show/Hide Map

Φράση:

Αποστολή

Καθαρισμός

Τι ξέρεις;

η όμορφη ψηλή γυναίκα του μανάβη περνά

Entities recognizer

Show/Hide Map

Φράση:

Αποστολή

Καθαρισμός

Τι ξέρεις;

ο μανάβης έχει όμορφη ψηλή γυναίκα

Entities recognizer

Show/Hide Map

Φράση:

Αποστολή

Καθαρισμός

Τι ξέρεις;

Λυπάμαι. Δε γνωρίζω

Σχήμα 39. Οι απαντήσεις σε διαφορετικά ερωτήματα αφότου δοθεί το κείμενο εισόδου: «Η όμορφη ψηλή γυναίκα του νέου μανάβη της γειτονιάς που περνά τώρα φοράει μοντέρνο πράσινο φόρεμα»

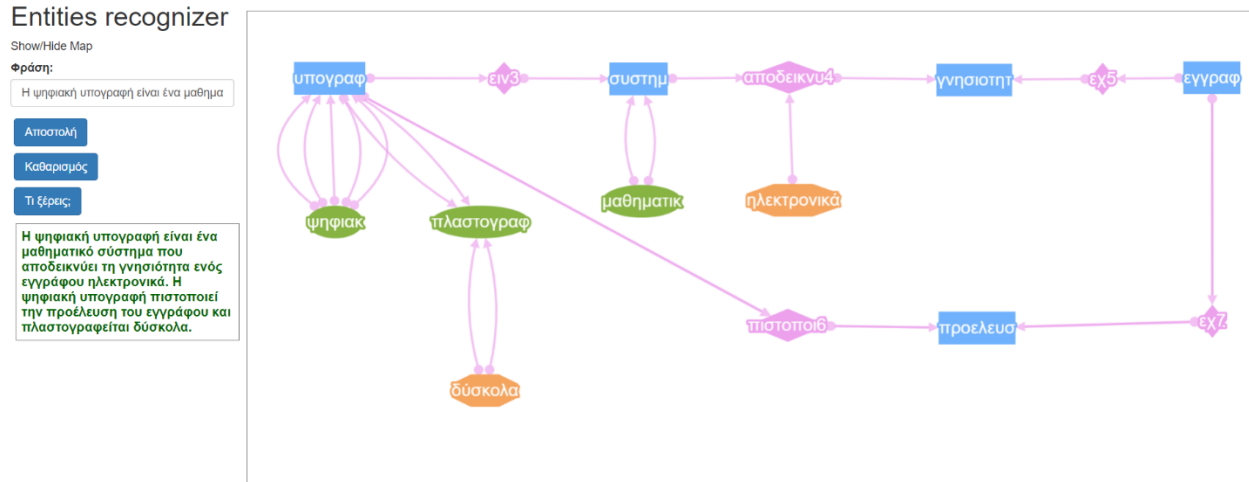
Παράδειγμα 4

Στο πλαίσιο κειμένου με τον τίτλο «Φράση» ο χρήστης πληκτρολογεί το κείμενο «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά. Η ψηφιακή υπογραφή πιστοποιεί την προέλευση του εγγράφου και πλαστογραφείται δύσκολα» και πατάει το κουμπί «Αποστολή». Τα περιεχόμενα της σελίδας που θα φορτώσει φαίνονται στο Σχήμα 40.

Στη συνέχεια ο χρήστης εισάγει ερωτήσεις στο σύστημα σχετικά με το κείμενο. Στο Σχήμα 41 φαίνεται η έξοδος της εφαρμογής σε δύο ερωτήματα.

- Στο ερώτημα «Τι είναι η ψηφιακή υπογραφή;» το σύστημα αποκρίνεται «Η ψηφιακή υπογραφή είναι μαθηματικό σύστημα».

- Στο ερώτημα «Ποιος πλαστογραφείται δύσκολα;» το σύστημα αποκρίνεται «Η ψηφιακή υπογραφή πλαστογραφείται δύσκολα».



Σχήμα 40. Τα περιεχόμενα της σελίδας της εφαρμογής για το κείμενο εισόδου: «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά. Η ψηφιακή υπογραφή πιστοποιεί την προέλευση του εγγράφου και πλαστογραφείται δύσκολα»

Entities recognizer

Show/Hide Map

Φράση:

Τι είναι η ψηφιακή υπογραφή;

Αποστολή

Καθαρισμός

Τι ξέρεις;

Τι είναι η ψηφιακή υπογραφή;

η ψηφιακή υπογραφή είναι μαθηματικό σύστημα

Entities recognizer

Show/Hide Map

Φράση:

Ποιος πλαστογραφείται δύσκολα;

Αποστολή

Καθαρισμός

Τι ξέρεις;

Ποιος πλαστογραφείται δύσκολα;

η ψηφιακή υπογραφή πλαστογραφείται δύσκολα

Σχήμα 41. Οι απαντήσεις σε διαφορετικά ερωτήματα αφότου δοθεί το κείμενο εισόδου: «Η ψηφιακή υπογραφή είναι ένα μαθηματικό σύστημα που αποδεικνύει τη γνησιότητα ενός εγγράφου ηλεκτρονικά. Η ψηφιακή υπογραφή πιστοποιεί την προέλευση του εγγράφου και πλαστογραφείται δύσκολα»

Παράδειγμα 5

Ο χρήστης πατάει το κουμπί «Τι ξέρεις;» και εμφανίζει στο παράθυρο τις πληροφορίες από τα λεξικά.

Entities recognizer

Show/Hide Map

Φράση:

τι ξέρεις

Αποστολή

Καθαρισμός

Τι ξέρεις

τι ξέρεις

Ρήματα αμετάβητα: περπατάω, μιλάω, χαιρετάω, περνάω, κουνάω, ταλαντώνω, ανοίγω, κλειδώνω, κινούμαι, πλαστογραφούμαι, προτείνωμαι, τρέχω, παίζω, κολυμπάω, γράφω

Ρήματα μεταβητικά: διαβάζω, δίνω, διδάσκω, παρακολουθώ, παρέχω, διαθέτω, περιλαμβάνω, αγαπάω, βοηθάω, αποδεικνύω, πιστοποιώ, δηλώνω, προκαλώ, χρησιμεύω, διευκολύνω, έχω, αναγνωρίζω, κερδίζω, απαιτώ, περιγράφω, αρχίζω, τελειώνω, εκτελώ, καθορίζω, χρησιμοποιούμαι, διεκπεραιώνω, καθορίζομαι, διδάσκομαι, αναφέρομαι, βάζω, αναπτύσσω, φέρω, παίζω, βλέπω, φοράω, κάνω, κλωτσώ, χρησιμοποιούω, είμαι

Ουσιαστικά αρσ: μαθητής, βαθμός, πυρετός, ιός, παραμυθιός, αριθμός, οριζόντιος, ιδανικός, χρόνος, υπάλληλος, ηλικίας, κάτοχος, μηχανικός, ψαράς, άντρας, βήχας, αποστολέας, βαρκάρης, χρήστης, καφές, παππούς, μανάβης, κατασκευαστής, πλανήτης, επιβάτης, καθηγητής, δρόμος, υπάλληλος, αδελφός, Γιάννης, Μαρακάκης, Νίκος, Γιώργος, Δίας, Κρόνος, Σεπτέμβριος

Ουσιαστικά θηλ: γυναίκα, δασκάλα, γραμμή, γνώση, διάγνωση, παραγωγή, αποφάση, παραμόρφωση, έναρξη, λήξη, ψήφος, είσοδος, μαμά, τράπεζα, ιλαρά, ικανότητα, δεξιάτητα, σταθερά, ημέρα, εβδομάδα, πάπια, ησυχία, λοίμωξη, κρυπτογραφία, πληροφορία, ιστορία, δημιουργία, διδασκαλία, εμπειρία, μπαταρία, ημερομηνία, μαθήτρια, τηλεόραση, τάξη, λέξη, εύθυνη, επαλήθευση, σχεδίαση, συντήρηση, εγκατάσταση, υποστήριξη, χώρα, αίσθηση, υπόσταση, γειτονιά, παράσταση, καταρροή, πληροφορική, ανάπτυξη, κατάθεση, κίνηση, περιστροφή, ταλάντωση, γέυση, φόρτιση, φυσική, γνώμη, αίτηση, μέθοδος, προέλευση, αναζήτηση, υπογραφή, αδελφή, απόδειξη, γνησιότητα, επαναφορά, εγκυρότητα, νοημοσύνη, παραμόρφωση, δύναμη, ενέργεια, Μαρία, Στέλλα

Ουσιαστικά ουδ: βιβλίο, κτίριο, συμβούλιο, ελατήριο, εργαστήριο, επιτόκιο, εγχειρίδιο, καπέλο, ιστορικό, ρούχο, έργο, παλτό, παιδί, κεφάλι, παπούτσι, φυτό, φωτοβολταϊκό, υλικό, φαγητό, κύπελο, νησί, έτος, μάθημα, θαύμα, ρεύμα, φόρεμα, γράψιμο, γάλα, πρόσωπο, σχολείο, χέρι, πόδι, λεμόνι, μάπι, κλειδί, μαλλί, σύστημα, αίτημα, σώμα, τμήμα, σύμπτωμα, μήνυμα, έγγραφο, εξάμηνο, χρώμα, μαθηματικά, γένος, κρέας

Επίθετα: μεγάλος, μεγάλη, μεγάλο, μεγαλύτερος, μεγαλύτερη, μεγαλύτερο, κοντός, κοντή, κοντό, προαπαιτούμενος, προαπαιτούμενη, προαπαιτούμενο, κινητικός, κινητική, κινητικό, μικρός, μικρή, μικρό, κόκκινος, κόκκινη, κόκκινο, πράσινος, πράσινη, πράσινο, κίτρινος, κίτρινη, κίτρινο, καστανός, καστανή, καστανό, νόστιμος, νόστιμη, νόστιμο, νέος, νέα, νέο, αξιόλογος, αξιόλογη, αξιόλογο, καστανός, καστανή, καστανό, άσπρος, άσπρη, άσπρο, εμπιστευτικός, εμπιστευτική, εμπιστευτικό, μαύρος, μαύρη, μαύρο, ψηλός, ψηλή, ψηλό, ανδρικός, ανδρική, ανδρικό, καθαρός, καθαρή, καθαρό, όμορφος,

7. Μελλοντικές επεκτάσεις

Στα πλαίσια της διατριβής έγινε η μελέτη, η σχεδίαση, η ανάπτυξη και οι δοκιμές του συστήματος ερωταποκρίσεων και όλων των υποσυστημάτων του και δημιουργήθηκε ένα ολοκληρωμένο σύστημα με συγκεκριμένες προδιαγραφές. Σκοπός κατά τη σχεδίαση του ήταν να είναι πλήρως λειτουργικό αλλά και επεκτάσιμο εκτός των πλαισίων αυτής της διατριβής μόνο με προσθήκες κώδικα και δεδομένων και με ελάχιστες ή μηδενικές αλλαγές στα δομικά του μέρη. Το σύστημα μπορεί να διαχειριστεί πληροφορία με μεγάλο βαθμό συντακτικής πολυπλοκότητας, όπως φαίνεται από τα παραδείγματα που παρατίθενται.

Το σύστημα ερωταποκρίσεων έχει συγκεκριμένες δυνατότητες και είναι ολοκληρωμένο. Ωστόσο το εύρος των γλωσσικών εκφράσεων είναι τεράστιο με πολλές λέξεις, συντακτικούς κανόνες και εξαιρέσεις. Εφόσον στα πλαίσια της διατριβής δεν ήταν δυνατόν να παραχθεί ένα σύστημα που να διαχειρίζεται με αξιοπιστία τον πλούτο των λέξεων και των εννοιών της Ελληνικής γλώσσας ούτε το σύνολο των κανόνων και των εξαιρέσεων, φροντίσαμε να σχεδιαστεί με γνώμονα την επεκτασιμότητα ώστε χωρίς σημαντικές δομικές αλλαγές, να μπορεί να διαχειρίζεται πολύ μεγαλύτερο όγκο πληροφοριών και να παρέχει απαντήσεις σε πολύπλοκα ερωτήματα προσθέτοντας ταυτόχρονα και νέους τύπους ερωτημάτων.

Η σχεδίαση του συστήματος επιτρέπει χωρίς δομικές αλλαγές να ενσωματώσει τις ακόλουθες δυνατότητες:

- Να επεκτείνει τα λεξικά του με την προσθήκη περισσότερων λέξεων στα αντίστοιχα αρχεία.
- Να επεκτείνει την αναγνώριση των ρημάτων σε όλους τους χρόνους, τις φωνές και τις εγκλίσεις .
- Να ενσωματώσει ξενόγλωσσα λεξικά για την αναγνώριση αγγλικών όρων σε ελληνικά κείμενα.
- Να ενσωματώσει λεξικά με συνώνυμα.
- Να επεκτείνει τα λεξικά του και να διαχειριστεί ανώμαλα ρήματα, επίθετα και ουσιαστικά.
- Να ενσωματώσει νέους κανόνες τονισμού για ανώμαλα επίθετα και ουσιαστικά.
- Να δεχτεί νέους κανόνες γραμματικής ώστε να καλύψει περιπτώσεις συντακτικού προτάσεων που στην τρέχουσα έκδοση δεν καλύπτονται.
- Να προστεθεί κώδικας για αυτόματη διόρθωση συνηθισμένων ορθογραφικών λαθών.
- Να επεκταθεί ώστε να μπορεί να εκπαιδευτεί από το χρήστη με εντολές σε φυσική γλώσσα και παραδείγματα (π.χ. «*Μάθε το φοιτητής όπως το μαθητής*»).
- Να επεκταθεί το εύρος των ερωτήσεων που απαντάει με ερωτήσεις τύπου Ποιον, Ποιου, Πόσοι.
- Να δημιουργεί απαντήσεις με συνάθροιση οντοτήτων π.χ. Πρόταση: «*Ο Γιώργος διαβάζει ένα βιβλίο. Η Μαρία διαβάζει ένα βιβλίο*», Ερ: «*Ποιος διαβάζει βιβλίο;*», Απ: «*Ο Γιώργος και η Μαρία διαβάζουν βιβλίο.*»
- Να επεκταθεί ώστε να ελέγχει σημασιολογικά την ορθότητα της αναπαράστασης.
- Να υποβάλλει διευκρινιστικές ερωτήσεις σε περίπτωση που οι απαντήσεις είναι περισσότερες από μία (π.χ. Ερ. Χρήστη: «*Τι διαβάζει ο μαθητής;*», Ερ. Συστήματος: «*Ο Γιάννης ή ο Νίκος;*», Απ. Χρήστη: «*Ο Νίκος.*», Απ. Συστήματος: «*Ο Νίκος διαβάζει φυσική.*»)

Επιπλέον η λειτουργία του συστήματος μέσα από web server του δίνει τη δυνατότητα, με χρήση κατάλληλου hardware με επαρκή υπολογιστική ισχύ, να εξυπηρετεί ταυτόχρονα πολλούς χρήστες. Επίσης η αναπαράσταση των οντοτήτων και των σχέσεων στη μορφή υποκείμενο – ρήμα – αντικείμενο

δίνει τη δυνατότητα επέκτασης της αναπαράστασης με τριπλέτες, γεγονός που επιτρέπει την ενσωμάτωση στο σύστημα του μοντέλου RDF. Το μοντέλο RDF χρησιμοποιείται για την εννοιολογική περιγραφή ή τη μοντελοποίηση των πληροφοριών που περιέχονται σε πόρους του διαδικτύου όπως και σε εφαρμογές διαχείρισης γνώσης.

Το σύστημα θεωρούμε ότι θα μπορούσε ιδανικά να λειτουργήσει παρέχοντας απαντήσεις σε χρήστες του διαδικτύου αντλώντας πληροφορίες από ηλεκτρονικά έγγραφα και ιστοσελίδες. Θα μπορούσε επίσης να λειτουργήσει ως διαλογικό σύστημα το οποίο θα παρέχει πληροφορίες σε χρήστες, συλλέγοντας ταυτόχρονα πληροφορίες για αυτούς ώστε οι απαντήσεις να μπορούν να είναι προσωποποιημένες (π.χ. προτάσεις για ταξίδια, ιατρικές συμβουλές κλπ). Επίσης θα μπορούσε να προστεθεί το πραγματολογικό πλαίσιο και το σύστημα να αντιλαμβάνεται ή να θυμάται πληροφορίες για ξεχωριστούς χρήστες χωρίς να διατυπώνονται σε κάθε ξεχωριστή πρόταση.

8. Συμπεράσματα

Ο αρχικός στόχος της μεταπτυχιακής διατριβής ήταν η ανάπτυξη ενός αυτοματοποιημένου τρόπου για τη μετατροπή κειμένου σε σημασιολογικές αναπαραστάσεις χωρίς επίβλεψη. Για το σκοπό αυτό χρησιμοποιήθηκαν επιλεγμένοι κανόνες του μοντέλου οντοτήτων – συσχετίσεων οι οποίοι είναι μια σειρά από κατευθυντήριες οδηγίες μετατροπής προδιαγραφών βάσεων δεδομένων σε σχήματα σχεσιακών βάσεων για την Αγγλική γλώσσα. Αφού επιλέχτηκαν οι κανόνες εκείνοι που μπορούσαν να εφαρμοστούν στην Ελληνική γλώσσα, προσθέσαμε επιπλέον συγκεκριμένους κανόνες οι οποίοι λαμβάνουν υπόψη την Ελληνική γραμματική. Στη συνέχεια σχεδιάστηκε και υλοποιήθηκε η μέθοδος αναπαράστασης της σημασιολογικής δομής των κειμένων ως γεγονότα των κατηγορημάτων `relModel/8`, `entity_types/3`, `relationship_type/3`, `attributes_of_entities/3`, `attributes_of_relationships/3` και `entities/3`.

Δημιουργήσαμε προγραμματιστικά όλα εκείνα τα γλωσσικά εργαλεία τα οποία είναι απαραίτητα για τη συντακτική ανάλυση κειμένων στην Ελληνική γλώσσα και εφαρμόσαμε τους κανόνες του Μοντέλου Οντοτήτων – Συσχετίσεων σε κείμενα τα οποία το σύστημα μετέτρεψε σε σημασιολογικά δέντρα. Σ αυτή τη φάση το θέμα που χρειάστηκε να ερευνηθεί ήταν κατά πόσο η αναπαράσταση αυτή απεικόνιζε σωστά τις πληροφορίες των αρχικών κειμένων και πόσο χρήσιμη μπορούσε να είναι στην πράξη μια τέτοια αναπαράσταση. Ένας τρόπος αξιολόγησης ήταν η αναζήτηση πληροφοριών μέσα σε αυτές τις σημασιολογικές δομές. Οι πληροφορίες που αναμένεται να επιστρέψει μια τέτοια αναζήτηση θα πρέπει να είναι σωστές και ακριβείς και οπωσδήποτε πιο αξιόπιστες από την απλή αναζήτηση λέξεων μέσα σε κείμενα.

Αφού το σύστημα είχε τη δυνατότητα να αναπαραστήσει κείμενα που περιείχαν προτάσεις, κατά συνέπεια μπορούσε να επεκταθεί ώστε να αναγνωρίζει και να αναπαριστά κείμενα που περιείχαν ερωτήσεις. Τα δέντρα σημασιολογικής αναπαράστασης των ερωτήσεων μπορούσαν να συγκριθούν με τα αντίστοιχα σημασιολογικά δέντρα των κειμένων στη βάση γνώση και, σε συνδυασμό με τον τύπο της ερώτησης, να εντοπιστούν στη βάση γνώσης οι πληροφορίες για τη σωστή απάντηση. Αναπτύξαμε έναν αλγόριθμο ο οποίος με βάση το μοντέλο της ερώτησης και συγκρίνοντάς το με τις αποθηκευμένες πληροφορίες από την αναπαράσταση των κειμένων στη βάση γνώσης, μπορεί να εντοπίζει τις οντότητες που αποτελούν απάντηση σε ερωτήσεις τύπου «Ποιος» και «Τι». Επιπλέον μπορεί να εντοπίζει τις ιδιότητες αυτών των οντοτήτων όπως και άλλες συνδεδεμένες οντότητες και να τις συμπεριλαμβάνει στις πληροφορίες της απάντησης.

Ο καλύτερος τρόπος για να παρασχεθούν οι πληροφορίες αυτές στο χρήστη που διατύπωσε την ερώτηση, είναι η ενσωμάτωσή τους σε μια πρόταση – απάντηση διατυπωμένη σωστά και σε συμφωνία με τον τύπο και τη διατύπωση της ερώτησης σε φυσική γλώσσα. Από τις δοκιμές που έγιναν, όπως φαίνεται και από τα σχετικά παραδείγματα, οι απαντήσεις προσεγγίζουν ή ταυτίζονται με τις απαντήσεις που θα έδινε ένα φυσικό πρόσωπο, στο οποίο θα είχαν τεθεί υπόψη τα κείμενα.

9. Βιβλιογραφικές Αναφορές

- [1] Al-Btoush, A. A.-S. (2015). Extracting Entity Relationship Diagram (ERD) from English Sentences. *International Journal of Database Theory and Application*, 8(2), 35-244.
- [2] Allam, A. M. (2012). The Question Answering Systems: A Survey. *International Journal of Research and Reviews in Information Sciences (IJRRIS)*, 2(3).
- [3] Al-Safadi, L. A. (2009). Natural Language Processing for Conceptual Modeling. *JDCTA*, 3(3), 47-59.
- [4] Amazon Comprehend. (2018, 1 28). Ανάκτηση από Natural Language Processing (NLP) and Machine Learning (ML): <https://aws.amazon.com/comprehend/>
- [5] Andrenucci, A .a. (2005). Automated Question Answering: Review of the Main Approaches. *IEEE*.
- [6] Androutopoulos, I. a. (2013). Generating natural language descriptions from OWL ontologies: the NaturalOWL system. *Journal of Artificial Intelligence Research*, 48, 671-715.
- [7] AUEB's NLP Group - Software. (2018, 1 28). Ανάκτηση από AUEB's NLP Group: http://nlp.cs.aueb.gr/software_gr.html
- [8] Badia, A. (2007). Question answering and database querying: Bridging the gap with generalized quantification. *Journal of Applied Logic*, 5(1), 3-19.
- [9] Barwise, J. (1977). *An introduction to first-order logic*. Elsevier.
- [10] Btoush, E. S. (2015). Generating ER diagrams from requirement specifications based on natural language processing. *International Journal of Database Theory and Application*, 8(2), 61-70.
- [11] Cambria, E., & White, B. (2014, 5). Jumping NLP Curves: A Review of Natural Language Processing Research. *IEEE Computational intelligence magazine*, 9(2), 48-57.
- [12] Ceccato, S. (1967). Correlational analysis and mechanical translation.
- [13] Chen, P. P.-S. (1976). The entity-relationship model—toward a unified view of data. *ACM Transactions on Database Systems*, 1(1), 9-36.
- [14] Chen, P. P.-S. (1983). English sentence structure and entity-relationship diagrams. *Information Sciences*, 29(2), 127-149.
- [15] Chomsky, N. (1956). Three models for the description of language. *IRE Transactions on information theory*, 113-124.
- [16] Cloud Natural Language. (2018, 1 28). Ανάκτηση από Google Cloud Platform: <https://cloud.google.com/natural-language/>
- [17] De Marneffe, M.-C. a. (2014). Universal Stanford dependencies: A cross-linguistic typology. *de2014universal*, (σσ. 4585-4592).
- [18] Dedhia, R. a. (2015). Techniques to automatically generate Entity Relationship Diagram. *International Journal of Innovations & Advancement in Computer Science (IJACS)*, 4(10), 68-73.
- [19] Definite Clause Grammars. (2018, 2 28). Ανάκτηση από SICStus Prolog Homepage: https://sicstus.sics.se/sicstus/docs/4.3.4/html/sicstus/ref_002dgru_002ddcg.html
- [20] Georgouli, A. (2015). Τεχνητή νοημοσύνη. Στο A. Georgouli, *Αναπαράσταση Γνώσης*. Hellenic Academic Libraries Link.
- [21] Gomez, F. a. (1999). A system for the semiautomatic generation of ER models from natural language specifications. *Data & Knowledge Engineering*, 1999, 57-81.

- [22] Google, Google Assistant - Your own personal. (2018, 1 24). Ανάκτηση από <https://assistant.google.com/>
- [23] ILSP NLP Web Services. (2018, 1 28). Ανάκτηση από <http://nlp.ilsp.gr/ws/>
- [24] iOS - Siri - Apple. (2018, 1 24). (Apple) Ανάκτηση από <http://www.apple.com/ios/siri/>
- [25] ISP NLP web service. (2018, 1 28). Ανάκτηση από Ινστιτούτο Επεξεργασίας του Λόγου: <http://nlp.ilsp.gr/ws/>
- [26] Kolomiyets, O. a.-F. (2011). A survey on question answering technology from an information retrieval perspective. Elsevier, 5412-5434.
- [27] Lampouras, G. (2015). PH.D. THESIS NATURAL LANGUAGE INTERACTION WITH SEMANTIC WEB ONTOLOGIES. DEPARTMENT OF INFORMATICS ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS.
- [28] Lampouras, G. a. (2013). Using integer linear programming for content selection, lexicalization, and aggregation to produce compact texts from OWL ontologies. Proceedings of the 14th European Workshop on Natural Language Generation, (σσ. 51-60).
- [29] Language Support | Google Cloud Natural Language API. (2018, 2 20). Ανάκτηση από Google Cloud Platform: <https://cloud.google.com/natural-language/docs/languages>
- [30] Linguistic Analysis API | Microsoft Azure. (2018, 2 28). Ανάκτηση από Microsoft Azure: <https://azure.microsoft.com/en-us/services/cognitive-services/linguistic-analysis-api/>
- [31] Marakakis, E., Kondylakis, H., & Aris, P. (2017). APANTISIS: A Greek Question-Answering System for Knowledge-Base Exploration. In Strategic Innovative Marketing (pp. 501-510). Springer, Cham.
- [32] McGuinness, D. L. (2004). OWL web ontology language overview. W3C recommendation.
- [33] Mendes, A. C. (2013). When the Answer comes into Question: Survey and open issues. Natural Language Engineering, 1-32.
- [34] Natural language processing. (2017). Wikipedia. Ανάκτηση October 19, 2017, από https://en.wikipedia.org/wiki/Natural_language_processing
- [35] Ntais, G. (2006). Development of a Stemmer for the Greek Language. Department of Computer and Systems Sciences Master Thesis at Stockholm University/Royal Institute of Technology.
- [36] NTUA Natural Language Processing Lab - ΕΜΠ Εργαστήριο Επεξεργασίας Φυσικής Γλώσσας. (2018, 1 28). Ανάκτηση από <http://glotta.ntua.gr/NLP/>
- [37] Omar, N. a. (2004). Heuristic-based entity-relationship modelling through natural language processing. Στο Artificial Intelligence and Cognitive Science Conference (AICS) (σσ. 302-313). Artificial Intelligence Association of Ireland (AIAI).
- [38] Omar, N. a. (2006). Semantic analysis in the automation of ER modelling through natural language processing. Στο Computing & Informatics, 2006. ICOCI'06. International Conference on (σσ. 1-5). IEEE.
- [39] Papageorgiou, H., Prokopidis, P., Giouli, V., Demiros, I., Konstantinidis, A., & Piperidis, S. (2002). Multi-level XML-based Corpus Annotation. In LREC.
- [40] Papakitsos, E. a. (2002). Modelling a Morpheme-based Lexicon for Modern Greek. Literary and linguistic computing, 475-490.
- [41] Pearl, J. (1985). Bayesian networks: A model of self-activated memory for evidential reasoning.

- [42] Peñas, A., Unger, C., Paliouras, G., & Kakadiaris, I. (2015, September). Overview of the CLEF Question Answering Track 2015. In International Conference of the Cross-Language Evaluation Forum for European Languages (pp. 539-544). Springer, Cham.
- [43] Poonam Gupta, V. G. (2012). A Survey of Text Question Answering Techniques. International Journal of Computer Applications.
- [44] Prokopidis, P., Georgantopoulos, B., & Papageorgiou, H. (2011). A suite of natural language processing tools for Greek. In The 10th international conference of Greek linguistics.
- [45] Shahbaz, M. a. (2011). Automatic generation of extended er diagram using natural language processing. Journal of American Science, 7(8).
- [46] Siri - Wikipedia. (2018, 1 24). (Wikipedia) Ανάκτηση από <https://en.wikipedia.org/wiki/Siri>
- [47] Sowa, J. F. (1987). Semantic networks. Encyclopedia of artificial intelligence.
- [48] Stanford Parser. (2018, 1 28). Ανάκτηση από <http://nlp.stanford.edu:8080parser/index.jsp>
- [49] Universal Dependency Relations. (2018, 2 20). Ανάκτηση από Universal Dependencies: <http://universaldependencies.org/u/dep/index.html>
- [50] Wang, M. (2006). A Survey of Answer Extraction Techniques in Factoid Question Answering. Association for Computational Linguistics, 1(1).
- [51] Watson (computer) - Wikipedia. (2018, 1 24). (Wikipedia) Ανάκτηση από [https://en.wikipedia.org/wiki/Watson_\(computer\)](https://en.wikipedia.org/wiki/Watson_(computer))
- [52] Wolfram|Alpha: Computational Knowledge Engine. (2018, 1 24). (Wolframalpha) Ανάκτηση από <https://www.wolframalpha.com/>
- [53] Yang, M. C., Duan, N., Zhou, M., & Rim, H. C. (2014). Joint relational embeddings for knowledge-based question answering. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 645-650).
- [54] Μαρακάκης, Μ. (2014). Prolog: Προγραμματισμός σε Λογική για Τεχνητή Νοημοσύνη. Εκδόσεις Νέων Τεχνολογιών.
- [55] Μοντέλο Οντοτήτων-Συσχετίσεων - Βικιπαίδεια. (2018, 1 28). (Βικιπαίδεια) Ανάκτηση 1 22, 2018, από Βικιπαίδεια: https://el.wikipedia.org/wiki/%CE%9C%CE%BF%CE%BD%CF%84%CE%AD%CE%BB%CE%BF_%CE%9F%CE%BD%CF%84%CE%BF%CF%84%CE%AE%CF%84%CF%89%CE%BD-%CE%A3%CF%85%CF%83%CF%87%CE%B5%CF%84%CE%AF%CF%83%CE%B5%CF%89%CE%BD
- [56] Οντολογία (Πληροφορική) - Wikipedia. (2018, 1 28). Ανάκτηση από Wikipedia: [https://el.wikipedia.org/wiki/%CE%9F%CE%BD%CF%84%CE%BF%CE%BB%CE%BF%CE%B3%CE%AF%CE%B1_\(%CF%80%CE%BB%CE%B7%CF%81%CE%BF%CF%86%CE%BF%CF%81%CE%B9%CE%BA%CE%AE\)](https://el.wikipedia.org/wiki/%CE%9F%CE%BD%CF%84%CE%BF%CE%BB%CE%BF%CE%B3%CE%AF%CE%B1_(%CF%80%CE%BB%CE%B7%CF%81%CE%BF%CF%86%CE%BF%CF%81%CE%B9%CE%BA%CE%AE))
- [57] Σακελλαρίου, Η. α. (2015). Επεξεργασία Φυσικής Γλώσσας και Γραμματικές Οριστικών Προτάσεων. Στο Η. α. Σακελλαρίου, Τεχνικές λογικού προγραμματισμού (σσ. 284-306). Kallipos publications.
- [58] Υπηρεσίες. (2018, 1 28). Ανάκτηση από Ινστιτούτο Επεξεργασίας του Λόγου: <http://www.ilsp.gr/el/services-products/services>