ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Αυτόματη Κατάταξη Ερωτήσεων Φυσικής Γλώσσας σε Κατηγορίες

Δημήτριος Μαυροειδής
Α.Μ. 3970050

Επιβλέπων Καθηγητής : Ίων Ανδρουτσόπουλος

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ, 2005

Περιεχόμενα

ΠΕΡΙΛΗΨΗ

Ένα σύστημα ερωταποκρίσεων φυσικής γλώσσας για συλλογές εγγράφων καλείται να εντοπίσει στα έγγραφα μιας συλλογής (π.χ. τα αρχεία μιας εφημερίδας ή τον Παγκόσμιο Ιστό) απαντήσεις σε ερωτήσεις που τίθενται σε φυσική γλώσσα. Για τη δημιουργία της απάντησης απαιτείται η επεξεργασία της ερώτησης, η ανάκτηση σχετικών εγγράφων από τη συλλογή με μια μηχανή ανάκτησης πληροφοριών, η επεξεργασία των εγγράφων, η αναζήτηση μέσα σε αυτά υποψηφίων απαντήσεων και η επιλογή της τελικής απάντησης. Η εργασία ασχολείται με το πρώτο στάδιο, την επεξεργασία της ερώτησης, και πιο συγκεκριμένα με την κατάταξη των ερωτήσεων σε κατηγορίες, ανάλογα με τον τύπο της απάντησης που απαιτούν (π.χ. όνομα προσώπου, τοποθεσίας, οργανισμού). Η κατηγορία της ερώτησης αποτελεί σημαντική πληροφορία, την οποία αξιοποιεί στη συνέχεια το στάδιο αναζήτησης υποψηφίων απαντήσεων.

Χρησιμοποιήθηκε κατά κύριο λόγο μηχανική μάθηση, πιο συγκεκριμένα Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines), και κατά δεύτερο λόγο γλωσσικά εργαλεία και πόροι, όπως ένα σύστημα Αναγνώρισης Κυρίων Ονομάτων (Named Entity Recognition) και ο ιεραρχικός θησαυρός WordNet. Το σύστημα που αναπτύχθηκε δέχεται ως είσοδο μια ερώτηση στα Αγγλικά και δίνει ως έξοδο την κατηγορία της ζητούμενης απάντησης. Πειραματικά αποτελέσματα δείχνουν ότι το σύστημα επιτυγχάνει περίπου το ίδιο καλά αποτελέσματα με εκείνα προηγούμενων ερευνητών.

1. ΕΙΣΑΓΩΓΗ

Η αναζήτηση πληροφοριών σε συλλογές εγγράφων (π.χ. αρχεία εφημερίδων ή ολόκληρο το Internet) γίνεται σήμερα κυρίως με την υποβολή λέξεων-κλειδιών σε μια μηχανή ανάκτησης πληροφοριών ή αναζήτησης ιστοσελίδων. Η μηχανή επιστρέφει μια λίστα με έγγραφα στα οποία βρήκε τις λέξεις-κλειδιά, επιστρατεύοντας και κριτήρια όπως το αν οι λέξεις-κλειδιά βρίσκονταν σε σημαντικά σημεία των εγγράφων (π.χ. τον τίτλο τους), αν είχαν μεγάλη συχνότητα εμφάνισης στα έγγραφα κλπ. Ο χρήστης στη συνέχεια πρέπει να εντοπίσει μόνος του στα έγγραφα που επεστράφησαν τις πληροφορίες που αναζητά, οι οποίες ενδέχεται να μην περιέχονται καν σε πολλά από αυτά τα έγγραφα. Τα συστήματα ερωταποκρίσεων (question-answering systems) φυσικής γλώσσας για συλλογές εγγράφων επιχειρούν να επεκτείνουν τις δυνατότητες των σημερινών μηχανών ανάκτησης πληροφοριών και αναζήτησης ιστοσελίδων. Επιτρέπουν στο χρήστη να εισάγει ερωτήσεις φυσικής γλώσσας (π.χ. «Ποιος σκότωσε τον Καποδίστρια;», «Ποια είναι η υψηλότερη βουνοκορφή στον κόσμο;», «Ποια η κοινή ονομασία του ακετυλοσαλικυλικού οξέος;», «Από τι αποτελείται μια γραφομηχανή;», «Τι είναι η caretta-caretta;»), αντί απλά λέξεις-κλειδιά, και προσπαθούν να επιστρέψουν ακριβείς απαντήσεις (π.χ. ονόματα προσώπων, εταιριών, ημερομηνίες κλπ. ή σύντομα αποσπάσματα) αντί για λίστες εγγράφων.

Ένα τέτοιο σύστημα ακολουθεί συνήθως τα εξής στάδια επεξεργασίας: επεξεργασία της ερώτησης, ανάκτηση σχετικών εγγράφων από τη συλλογή με μια μηχανή ανάκτησης πληροφοριών (ή αναζήτησης ιστοσελίδων) χρησιμοποιώντας τους όρους της ερώτησης ως λέξεις-κλειδιά, επεξεργασία των εγγράφων, αναζήτηση μέσα σε αυτά υποψηφίων απαντήσεων και επιλογή της τελικής απάντησης. Η εργασία ασχολείται με το πρώτο στάδιο, την επεξεργασία της ερώτησης, και πιο συγκεκριμένα με το υπο-στάδιο της κατάταξης των ερωτήσεων σε κατηγορίες, ανάλογα με τον τύπο της απάντησης που απαιτούν (π.χ. όνομα προσώπου, τοποθεσίας, οργανισμού). Η κατηγορία της ερώτησης αποτελεί σημαντική πληροφορία, την οποία αξιοποιεί στη συνέχεια το στάδιο αναζήτησης υποψηφίων απαντήσεων. Χρησιμοποιήσαμε κατά κύριο λόγο μηχανική μάθηση, πιο συγκεκριμένα Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines), και κατά δεύτερο λόγο γλωσσικά εργαλεία και πόρους, όπως ένα σύστημα Αναγνώρισης Κυρίων Ονομάτων (Named Entity Recognition) και τον ιεραρχικό θησαυρό WordNet. Το σύστημα που αναπτύχθηκε δέχεται ως είσοδο μια ερώτηση στα Αγγλικά και δίνει ως έξοδο την κατηγορία της ζητούμενης απάντησης.

Ο πρώτος στόχος της εργασίας ήταν να επαληθεύσουμε τα αποτελέσματα των μεθόδων που δημοσιεύθηκαν σε δύο προηγούμενες εργασίες των Zhang & Lee [39] και Li & Roth [18]. Ένας δεύτερος στόχος ήταν να βελτιώσουμε τις μεθόδους αυτές, ώστε να πετύχουμε καλύτερα αποτελέσματα. Όσον αφορά τον πρώτο στόχο, επετεύχθησαν αποτελέσματα ελαφρά υποδεέστερα από εκείνα των προηγούμενων εργασιών, γεγονός που οφείλεται κατά πάσα πιθανότητα στη χρήση διαφορετικών υλοποιήσεων αλγορίθμων μηχανικής μάθησης. Όσον αφορά το δεύτερο στόχο, αξιολογήσαμε πειραματικά διάφορες παραλλαγές των προηγούμενων μεθόδων, προσπαθώντας να τις βελτιώσουμε, αλλά δεν καταφέραμε να ξεπεράσουμε τα ποσοστά επιτυχίας των προηγούμενων εργασιών.

Η εργασία ξεκινά με μια σύντομη περιγραφή των συστημάτων ερωταποκρίσεων (κεφάλαιο 2). Στη συνέχεια (κεφάλαιο 3) γίνεται μια εισαγωγή στη μηχανική μάθηση, καθώς και τον αλγόριθμο που χρησιμοποιείται στην εργασία – τις Μηχανές Διανυσμάτων Υποστήριξης. Στο τέταρτο κεφάλαιο περιγράφονται αναλυτικά οι μέθοδοι που διερευνήσαμε και τα αποτελέσματα των πειραμάτων μας.

Το πέμπτο κεφάλαιο αναφέρει τα συμπεράσματά μας και πιθανές βελτιώσεις που επιδέχεται η προσέγγισή μας.

2. ΣΥΣΤΗΜΑΤΑ ΕΡΩΤΑΠΟΚΡΙΣΕΩΝ

2.1. TREC

Με σκοπό τη δημιουργία ολοκληρωμένων συστημάτων ανάκτησης πληροφοριών, δημιουργήθηκε με πρωτοβουλία του NIST (National Institute of Standards and Technology) και του Υπουργείου Αμύνης των ΗΠΑ το Συνέδριο Ανάκτησης Πληροφοριών (TREC – Text REtrieval Conference [37], βλ. http://trec.nist.gov). Το TREC έχει σκοπό την υποστήριξη της έρευνας στο αντικείμενο της ανάκτησης πληροφοριών, παρέχοντας την απαραίτητη υποδομή για μεγάλης κλίμακας πειραματική αξιολόγηση σχετικών μεθόδων. Το συνέδριο διοργανώνει κάθε χρόνο διαγωνισμούς συστημάτων ανάκτησης πληροφοριών. Οι διαγωνισμοί ξεκίνησαν το 1992 και συνεχίζεται έκτοτε με συνεχώς περισσότερους συμμετέχοντες.

Οι διαγωνισμοί του TREC διευρύνονται συνεχώς με νέους τομείς (tracks). Το 1999 (TREC-8) δημιουργήθηκε ο τομέας για τα συστήματα ερωταποκρίσεων. Από τότε έχουν παρουσιαστεί πολλά συστήματα ερωταποκρίσεων και έχουν προταθεί διαφορετικές προσεγγίσεις και αρχιτεκτονικές. Σε γενικές γραμμές, όμως, η πλειοψηφία των συστημάτων ερωταποκρίσεων ακολουθεί τη γενική αρχιτεκτονική που περιγράφεται παρακάτω.

2.2. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΩΝ ΕΡΩΤΑΠΟΚΡΙΣΕΩΝ

Ένα σύστημα ερωταποκρίσεων ακολουθεί συνήθως τρία στάδια επεξεργασίας:

α) Το πρώτο στάδιο είναι η **επεξεργασία της ερώτησης**. Αυτό είναι ένα απαραίτητο βήμα για να καταλάβουμε τι ακριβώς ζητάει η ερώτηση. Η επεξεργασία της ερώτησης περιλαμβάνει, μεταξύ άλλων, τον καθορισμό του τύπου της απάντησης που αναμένεται. Γνωρίζοντας, παραδείγματος χάριν, ότι αναζητάμε ένα όνομα προσώπου, στενεύουμε κατά πολύ τον ορίζοντα αναζήτησης της απάντησης, αφού μπορούμε να περιοριστούμε σε τμήματα εγγράφων που αποτελούν ή περιέχουν ονόματα προσώπων. Μπορούμε επίσης, κατά την ανάλυση της ερώτησης, να πάρουμε και πληροφορίες άλλου τύπου, όπως τους όρους της ερώτησης, το συντακτικό της δέντρο, συνώνυμα των όρων της ερώτησης κλπ., τις οποίες αξιοποιούν τα μετέπειτα στάδια.

β) Το δεύτερο στάδιο διεξάγει **αναζήτηση για υποψήφιες απαντήσεις** στη συλλογή εγγράφων. Το στάδιο αυτό αποτελείται συχνά από τρία υπο-στάδια. Το πρώτο υποβάλλει τους όρους της ερώτησης (πιθανώς επεκτείνοντάς τους με συνώνυμα κλπ.) σε μια μηχανή ανάκτησης πληροφοριών ή αναζήτησης ιστοσελίδων, η οποία επιστρέφει σχετικά έγγραφα. Το δεύτερο επεξεργάζεται τα έγγραφα που επεστράφησαν (π.χ. διαχωρισμός σε περιόδους, συντακτική ανάλυση, εντοπισμός κυρίων ονομάτων κλπ.). Το τρίτο εντοπίζει υποψήφιες απαντήσεις μέσα στα έγγραφα που επεστράφησαν (π.χ. προτάσεις που περιέχουν ονόματα προσώπων αν η ερώτηση ζητά όνομα προσώπου).

γ) Στο τρίτο στάδιο γίνεται η **επιλογή της υποψήφιας απάντησης ή κάποιου μέρους της** που θα αποτελέσει την τελική απάντηση του συστήματος στη δοθείσα ερώτηση. Αυτή η επιλογή γίνεται συνήθως χρησιμοποιώντας αλγορίθμους μηχανικής μάθησης, αφού οι υποψήφιες απαντήσεις παρασταθούν ως διανύσματα ιδιοτήτων που περιλαμβάνουν πληροφορίες όπως το ποσοστό των όρων της ερώτησης που περιέχει

η υποψήφια απάντηση, το κατά πόσον το συντακτικό δέντρο της υποψήφιας απάντησης ταιριάζει με εκείνο της ερώτησης κλπ.

2.3. ΚΑΤΑΤΑΞΗ ΕΡΩΤΗΣΕΩΝ ΣΕ ΚΑΤΗΓΟΡΙΕΣ

Σκοπός της παρούσης εργασίας ήταν να κατασκευαστεί ένα σύστημα για το υπο-στάδιο που προσδιορίζει τον τύπο της αναμενόμενης απάντησης κάθε ερώτησης, δηλαδή ένα σύστημα που θα δέχεται ως είσοδο ερωτήσεις και θα τις ταξινομεί σε κατηγορίες, ανάλογα με τους τύπους των απαντήσεων που ζητούν. Για την κατάταξη των ερωτήσεων σε κατηγορίες έχουν προταθεί πολλοί τρόποι από ερευνητές της περιοχής ([8], [10], [12], [18], [35], [36], [39]). Οι περισσότεροι κατασκευάζουν κανόνες που εξετάζουν τις λέξεις των ερωτήσεων. Κυρίως λαμβάνεται υπόψιν η πρώτη λέξη· αυτό είναι εύλογο, καθώς, παραδείγματος χάριν, όταν ζητάμε να μάθουμε το όνομα κάποιου προσώπου η ερώτηση συχνά ξεκινά με «Ποιος», «Ποια», «Ποιο» κτλ., ενώ ερωτήσεις που ξεκινούν με «Πού», «Πότε», «Πώς», «Γιατί» κατά κανόνα ζητούν τοποθεσία, χρόνο, τρόπο και αιτία, αντίστοιχα. Υπάρχουν, όμως, ερωτήσεις των οποίων το ζητούμενο δεν είναι τόσο εύκολο να διαπιστωθεί. Αυτές μπορεί να ξεκινούν από «Τι», «Ονόμασε», «Από τι» κ.α. Παραδείγματα είναι τα εξής: «Τι είναι το αυγό του Κολόμβου;», «Ονόμασε τους 7 σοφούς της αρχαιότητας.», «Από τι φτιάχνεται η ταραμοσαλάτα;». Στις ερωτήσεις αυτές υπάρχει μεγάλη δυσκολία εντοπισμού της κατηγορίας τους – τι, δηλαδή, ζητάνε ως απάντηση. Έχουν προταθεί διάφοροι τρόποι για να ξεπεραστεί αυτή η δυσκολία. Ένας από αυτούς είναι η χρήση επιβλεπόμενης μηχανικής μάθησης (βλέπε κεφάλαιο 3).

Στην περίπτωση που χρησιμοποιείται επιβλεπόμενη μηχανική μάθηση, πρέπει πρώτα να κατασκευάσουμε κατηγορίες ερωτήσεων. Επιλέξαμε τις ίδιες κατηγορίες που χρησιμοποίησαν οι Zhang & Lee [39] και Li & Roth [18], γεγονός που μας βοηθά στη σύγκριση των αποτελεσμάτων μας με εκείνα που έχουν δημοσιεύσει οι συγκεκριμένοι ερευνητές. Στη συνέχεια, πρέπει να κατασκευαστεί ένα σύνολο ερωτήσεων που θα αποτελέσουν τα δεδομένα εκπαίδευσης του αλγορίθμου μηχανικής μάθησης· κάθε μία από αυτές τις ερωτήσεις πρέπει να αντιστοιχηθεί χειρωνακτικά με την κατηγορία (ή τις κατηγορίες) στην οποία ανήκει. Ο αλγόριθμος μηχανικής μάθησης εκπαιδεύεται στα δεδομένα εκπαίδευσης, και στη συνέχεια χρησιμοποιείται για να καταταχθούν σε κατηγορίες νέες ερωτήσεις των οποίων δεν γνωρίζουμε τις σωστές κατηγορίες. Η αξιοπιστία του συστήματος, δηλαδή το ποσοστό των ερωτήσεων που κατατάσσεται στη σωστή κατηγορία, υπολογίζεται με τη βοήθεια μιας ξεχωριστής συλλογής ερωτήσεων αξιολόγησης.

2.4. ΠΡΟΗΓΟΥΜΕΝΕΣ ΠΡΟΣΕΓΓΙΣΕΙΣ

Στο παρελθόν υπήρξαν ορισμένοι ερευνητές οι οποίοι χρησιμοποίησαν μηχανική μάθηση για την κατάταξη ερωτήσεων, με πολύ καλά αποτελέσματα. Μεταξύ αυτών αποφασίσαμε να στηριχθούμε στις εργασίες των Zhang & Lee [39] και Li & Roth [18], οι οποίες είχαν τα καλύτερα αποτελέσματα. Τα αποτελέσματα των εργασιών αυτών συγκρίνονται με τα αποτελέσματα του δικού μας συστήματος στο κεφάλαιο 4.

Χρονικά προηγήθηκε η εργασία των Li & Roth [18], η οποία χρησιμοποίησε τον αλγόριθμο μηχανικής μάθησης SNoW και πέτυχε ποσοστό επιτυχίας 84,2% κατά την κατάταξη των ερωτήσεων. Ακολούθησαν οι Zhang & Lee [39], οι οποίοι ακολούθησαν την ίδια μεθοδολογία με τους προηγούμενους, με τη διαφορά ότι

χρησιμοποίησαν ΜΔΥ (Μηχανή Διανυσμάτων Υποστήριξης). Χρησιμοποιώντας μια ΜΔΥ με έναν πυρήνα που μπορεί να εκμεταλλεύεται και πληροφορίες για τη συντακτική δομή της ερώτησης, πέτυχαν ποσοστά επιτυχίας μέχρι και 80,2%.

2.5. Η ΔΙΚΗ ΜΑΣ ΠΡΟΣΕΓΓΙΣΗ

Έχοντας υπόψη τα παραπάνω καλά αποτελέσματα, πρώτος στόχος της εργασίας ήταν να τα επιβεβαιώσει. Ακολουθήσαμε με προσοχή τα βήματα των παραπάνω δύο εργασιών και καταλήξαμε σε παρόμοια αποτελέσματα. Τα ποσοστά μας ήταν ελαφρώς χαμηλότερα, γεγονός που οφείλεται – κατά πάσα πιθανότητα – στην υλοποίηση και τις παραμέτρους της ΜΔΥ που χρησιμοποιήσαμε. Η υλοποίηση της ΜΔΥ στην οποία στηριχθήκαμε περιέχεται στο Weka[1], ένα γενικό εργαλείο μηχανικής μάθησης, ενώ χρησιμοποιήσαμε τις προεπιλεγμένες τιμές των παραμέτρων. Έγιναν πειράματα με πολυωνυμικό πυρήνα της ΜΔΥ και διαφορετικές τιμές για το βαθμό του πυρήνα. Η ΜΔΥ μάς έδινε καλύτερα αποτελέσματα για πολυωνυμικό πυρήνα $2^{ου}$ βαθμού, αλλά δεν μπορούσε να διαχειριστεί μεγάλα σύνολα εκπαίδευσης (πάνω από 300 ερωτήσεις) λόγω μεγάλων απαιτήσεων σε μνήμη. Έτσι περιοριστήκαμε σε πυρήνα $1^{ου}$ βαθμού (γραμμική ΜΔΥ).

Μεταξύ των βελτιώσεων που διερευνήσαμε ήταν η χρήση του εργαλείου Αναγνώρισης Κυρίων Ονομάτων (Named Entity Recognizer) που περιλαμβάνεται στο ελεύθερα διαθέσιμο σύστημα GATE[2], ένα γενικό σύστημα ανάπτυξης εφαρμογών φυσικής γλώσσας. Με τη βοήθεια του εργαλείου αυτού, αντικαταστάθηκαν τα κύρια ονόματα που περιέχονταν στις ερωτήσεις με τους τύπους των ονομάτων, ώστε να δοθεί στη ΜΔΥ η δυνατότητα μεγαλύτερης γενίκευσης. Για παράδειγμα, τα ονόματα «United Nations» και «Marie Curie» αντικαταστάθηκαν από τους τύπους «Organization» και «Person» αντίστοιχα.

Πειραματιστήκαμε, επίσης, με τη χρήση του ιεραρχικού θησαυρού λέξεων WordNet[3] [7]. Το WordNet παρέχει, μεταξύ άλλων, πληροφορίες για τα συνώνυμα, υπερώνυμα, υπώνυμα κλπ. των διαφόρων λέξεων και οι πληροφορίες αυτές μπορούν να χρησιμοποιηθούν, παραδείγματος χάριν, για την αντικατάσταση λέξεων της ερώτησης με λέξεις γενικότερης σημασίας, κατ' αντιστοιχία με την αντικατάσταση κυρίων ονομάτων με τους τύπους τους, ώστε και πάλι να δοθεί στη ΜΔΥ η δυνατότητα μεγαλύτερης γενίκευσης.

[1] http://www.cs.waikato.ac.nz/~ml/weka/
[2] http://gate.ac.uk/
[3] http://wordnet.princeton.edu/

3. ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

3.1. ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ ΚΑΙ ΚΑΤΑΤΑΞΗ ΕΡΩΤΗΣΕΩΝ

Η Μηχανική Μάθηση αποτελεί έναν κλάδο της Τεχνητής Νοημοσύνης ο οποίος αναπτύσσει τεχνικές που επιτρέπουν σε υπολογιστικά συστήματα να «μαθαίνουν», δηλαδή να αυτοβελτιώνονται μέσω της αξιοποίησης εμπειρικών δεδομένων του παρελθόντος. Ένας πιο ακριβής ορισμός ανήκει στον Mitchell [22] :

> *Ένα πρόγραμμα μαθαίνει από την εμπειρία που αποκτά κατά την εκτέλεση ενός συνόλου διεργασιών, εφόσον η απόδοσή του βελτιώνεται με την αξιοποίηση της εμπειρίας αυτής.*

Συνήθως η εμπειρία αναφέρεται και ως «Δεδομένα Εκπαίδευσης». Στην παρούσα εργασία ασχολούμαστε με Επιβλεπόμενη Μηχανική Μάθηση, όπου τα δεδομένα εκπαίδευσης έχουν συνήθως τη μορφή διανυσμάτων και κάθε διάνυσμα αποτελείται από μια τιμή-στόχο και πολλές τιμές ιδιοτήτων. Ας θεωρήσουμε ένα απλοϊκό παράδειγμα:

Έστω ότι διαθέτουμε έναν μετρητή ταχύτητας ανέμου, ένα βαρόμετρο και ένα θερμόμετρο. Θέλουμε να φτιάξουμε ένα υποτυπώδες σύστημα πρόβλεψης του καιρού. Κάθε μέρα στις 18:00, παίρνουμε μετρήσεις από τα τρία όργανα που διαθέτουμε και καταγράφουμε τις τιμές που αυτά δείχνουν. Την επόμενη μέρα το πρωί καταγράφουμε αν έχει λιακάδα ή συννεφιά. Εισάγουμε τις τέσσερις αυτές τιμές σε ένα διάνυσμα. Ας υποθέσουμε ότι κάνουμε μετρήσεις για ένα έτος. Στο τέλος του έτους θα έχουμε 365 διανύσματα της μορφής:

$$\vec{t} = \{[ταχύτητα\_ανέμου],[ατμοσφαιρική\_πίεση],[θερμοκρασία],[αυριανός\_καιρός]\}$$

Στα διανύσματα αυτά, οι τρεις πρώτες τιμές αποτελούν τιμές των ιδιοτήτων «ταχύτητα ανέμου», «ατμοσφαιρική πίεση» και «θερμοκρασία», και η τελευταία τιμή είναι η τιμή-στόχος. Με διαφορετική ορολογία, θα μπορούσαμε να πούμε ότι οι τρεις πρώτες τιμές αποτελούν τα ορίσματα μιας συνάρτησης και η τελευταία τιμή το αποτέλεσμά της. Παραδείγματα τέτοιων διανυσμάτων είναι τα $\vec{u}$ και $\vec{v}$ παρακάτω:

$$\vec{u} = \{6, 910, 14, συννεφιά\}$$
$$\vec{v} = \{3, 1025, 25, λιακάδα\}$$

Τα 365 διανύσματα εισάγονται σε ένα σύστημα μηχανικής μάθησης ως δεδομένα εκπαίδευσης. Το σύστημα εκπαιδεύεται με βάση αυτά τα διανύσματα σύμφωνα με κάποιον αλγόριθμο μάθησης. Στη συνέχεια, δίνοντας στο σύστημα οποιεσδήποτε τιμές ταχύτητας ανέμου, ατμοσφαιρικής πίεσης και θερμοκρασίας, πρέπει να μπορεί να μας δώσει πρόβλεψη για τον αυριανό καιρό (δηλαδή μία από τις τιμές «λιακάδα» ή «συννεφιά»).

Η τιμή-στόχος, στο παραπάνω παράδειγμα, είναι δυαδική («λιακάδα» - «συννεφιά»). Υπάρχουν περιπτώσεις όπου η τιμή-στόχος μπορεί να έχει πάνω από δύο εκδοχές. Στο ίδιο παράδειγμα – χάριν εμπλουτισμού των προβλέψεών μας – μπορούμε να έχουμε επιπλέον «βροχή», «καταιγίδα», «καύσωνας», κ.ο.κ. Κάποιοι αλγόριθμοι μηχανικής μάθησης είναι εκ κατασκευής δυαδικοί, μπορούν, δηλαδή, να αντιμετωπίσουν μόνο 2 τιμές-στόχους. Συχνά, όμως, με κατάλληλες μετατροπές μπορούν να αντιμετωπίσουν πάνω από 2 τιμές-στόχους.

Ας επιστρέψουμε τώρα στο πρόβλημα της κατάταξης ερωτήσεων σε κατηγορίες. Η τιμή-στόχος είναι η κατηγορία στην οποίαν ανήκει η ερώτηση. Στην απλούστερη περίπτωση, κάθε μία από τις ιδιότητες δείχνει αν εμφανίζεται ή όχι στην ερώτηση κάποια συγκεκριμένη λέξη. Έτσι, ένα διάνυσμα εκπαίδευσης που θα εισαγόταν στο σύστημα μηχανικής μάθησης θα μπορούσε να μοιάζει με το παρακάτω:

$$\vec{v} = \{0,0,1,0,1,1,1,0,0,0,27\}$$

Θεωρούμε εδώ πως έχουμε 10 συνολικά ιδιότητες, που αντιστοιχούν κατά σειρά στις λέξεις «και», «αν», «πώς», «αλλά», «φτιάχνονται», «τα», «γαριδάκια», «γιατί», «τι», «από». Το παραπάνω διάνυσμα μας λέει ότι η ερώτηση περιέχει τις λέξεις: «πώς», «φτιάχνονται», «τα» και «γαριδάκια», ενώ δεν περιέχει καμία άλλη λέξη που αντιστοιχεί σε ιδιότητα. Επίσης, μας λέει ότι ο τύπος της ερώτησης είναι «27». Το «27» αντιπροσωπεύει μια κατηγορία ερώτησης, όπως «Πρόσωπο», «Τοποθεσία», «Αντικείμενο» κ.τ.λ.

Στην εργασία των Li & Roth [18] χρησιμοποιείται ο αλγόριθμος μηχανικής μάθησης SNoW [3], με τη βοήθεια του οποίου αναπτύχθηκε ένας ταξινομητής που μπορεί να κατατάξει κάθε ερώτηση σε μια από τις 50 κατηγορίες μιας ιεραρχικής ταξινομίας ερωτήσεων. Στα πειράματά τους, οι Li και Roth χρησιμοποίησαν ως δεδομένα εκπαίδευσης σύνολα των 1000, 2000, 3000, 4000, και 5452 ερωτήσεων και ένα σύνολο 500 ερωτήσεων ως δεδομένα αξιολόγησης. Για τις 5452 ερωτήσεις, το ποσοστό ορθής κατάταξης νέων ερωτήσεων ήταν 84,2%.

Η εργασία των Zhang & Lee [39] συγκρίνει διαφόρους αλγορίθμους μηχανικής μάθησης (Nearest Neighbor [38], Naïve Bayes [19], C4.5 [31], SNoW [3]). Επίσης, χρησιμοποιεί ως ιδιότητες εκτός από απλές λέξεις και ακολουθίες λέξεων (n-grams). Η εργασία επικεντρώνεται στις Μηχανές Διανυσμάτων Υποστήριξης και δείχνει πως οι ΜΔΥ δίνουν καλύτερα αποτελέσματα από τους άλλους αλγορίθμους. Συγκεκριμένα, με το ίδιο σύνολο των 5452 ερωτήσεων εκπαίδευσης και 500 ερωτήσεων αξιολόγησης που χρησιμοποίησαν οι Li & Roth [18], το ποσοστό ορθής κατάταξης νέων ερωτήσεων της ΜΔΥ ήταν 80,2%, ενώ του SNoW ήταν 74%. Η διαφορά που προκύπτει στα ποσοστά επιτυχίας του SNoW ανάμεσα στις δύο εργασίες, οφείλεται – κατά πάσα πιθανότητα – στη χρήση κατωφλίου ίσου με 10 από τους Zhang & Lee [39] (βλ. ενότητα 4.2 παρακάτω).

3.2. ΜΗΧΑΝΕΣ ΔΙΑΝΥΣΜΑΤΩΝ ΥΠΟΣΤΗΡΙΞΗΣ

Οι ΜΔΥ είναι μια οικογένεια μεθόδων επιβλεπόμενης μάθησης, που μπορούν, μεταξύ άλλων, να χρησιμοποιηθούν στην κατάταξη διαφόρων αντικειμένων σε κατηγορίες. Οι ΜΔΥ μπορούν γενικά να διαχωριστούν σε δυαδικές (διταξικές – binary) και πολυταξικές (multiclass). Οι πολυταξικές ΜΔΥ μπορούν, στην απλούστερη περίπτωση, να υλοποιηθούν ως διαδοχική εφαρμογή δυαδικών ΜΔΥ.

Οι δυαδικές ΜΔΥ δημιουργούν ένα υπερεπίπεδο μεγίστου περιθωρίου, το οποίο βρίσκεται σε ένα διαφοροποιημένο από τον αρχικό διανυσματικό χώρο, που προκύπτει με τη χρήση ενός εν γένει μη γραμμικού μετασχηματισμού. Δεδομένων διανυσμάτων εκπαίδευσης που έχουν ταξινομηθεί σε μία από τις δύο κατηγορίες 0 ή 1, το υπερεπίπεδο επιχειρεί να διαχωρίσει στο νέο χώρο τα διανύσματα εκπαίδευσης έτσι ώστε η απόσταση του υπερεπιπέδου από τα κοντινότερα διανύσματα εκπαίδευσης (το περιθώριο) να είναι η μέγιστη δυνατή. Η χρήση του υπερεπιπέδου μεγίστου περιθωρίου στηρίζεται στη Θεωρία Στατιστικής Μάθησης, που προσφέρει ένα όριο πιθανοτικού σφάλματος ελέγχου το οποίο ελαχιστοποιείται όταν το

περιθώριο μεγιστοποιείται. Η εξεύρεση του βέλτιστου υπερεπιπέδου γίνεται με την επίλυση ενός προβλήματος βελτιστοποίησης με μεθόδους μαθηματικού προγραμματισμού. Για περισσότερες πληροφορίες, ανατρέξτε στο βιβλίο των Cristianini και Shawe-Taylor [5].

4. ΜΕΘΟΔΟΙ ΚΑΙ ΠΕΙΡΑΜΑΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΗΣ ΕΡΓΑΣΙΑΣ

4.1. ΕΙΣΑΓΩΓΗ

Ο πρώτος στόχος ήταν η επανάληψη των πειραμάτων των εργασιών των Zhang & Lee [39], Li & Roth [18] και η επίτευξη παραπλησίων ποσοστών επιτυχίας στην κατάταξη των ερωτήσεων σε κατηγορίες. Χρησιμοποιήσαμε στα πειράματά μας το ίδιο σύνολο ερωτήσεων που χρησιμοποίησαν οι Li & Roth [18], το οποίο υιοθέτησαν και οι Zhang & Lee [39], ώστε τα αποτελέσματά μας να είναι συγκρίσιμα. Το σύνολο αυτό αποτελείται από 5452 ερωτήσεις εκπαίδευσης και 500 ερωτήσεις αξιολόγησης. (Τα σύνολα των ερωτήσεων βρίσκονται στο διαδικτυακό τόπο: http://L2R.cs.uiuc.edu/~cogcomp/ [18].) Για το σύνολο εκπαίδευσης, διατίθενται 5 υποσύνολα των 1000, 2000, 3000, 4000 και 5452 ερωτήσεων, που χρησιμοποιούνται σε πειράματα με μεταβλητό μέγεθος δεδομένων εκπαίδευσης. Οι 500 ερωτήσεις αξιολόγησης προέρχονται από το TREC-10.

Χρησιμοποιήσαμε, επίσης, τις ίδιες κατηγορίες ερωτήσεων με εκείνες των Li & Roth [18]. Οι κατηγορίες είναι δύο επιπέδων, όπως φαίνεται στον παρακάτω πίνακα. Στο πρώτο επίπεδο υπάρχουν γενικές κατηγορίες και στο δεύτερο πιο ειδικές.

Κυρίως κατηγορία	**Υποκατηγορία**	**#**
ABBREV.	Abb	1
	Exp	2
ENTITY	Animal	**3**
	Body	4
	Colour	5
	Creative	6
	Currency	7
	Dis.med.	8
	Event	9
	Food	10
	Instrument	11
	Lang	12
	Letter	13
	Other	14
	Plant	15
	Product	16
	Religion	17
	Sport	18
	Substance	19
	Symbol	20
	Technique	21
	Term	22
	Vehicle	23
	Word	24
DESCRIPTION	Definition	25
	Description	26
	Manner	27
	Reason	28
HUMAN	Group	29
	Individual	30
	Title	31
	Description	32
LOCATION	City	33

	Country	34
	Mountain	35
	Other	36
	State	37
NUMERIC	Code	38
	Count	39
	Date	40
	Distance	41
	Money	42
	Order	43
	Other	44
	Period	45
	Percent	46
	Speed	47
	Temp	48
	Size	49
	Weight	50

Στα πειράματά μας λάβαμε υπόψη μας μόνο τα ποσοστά επιτυχίας για τις ειδικές κατηγορίες (του $2^{ου}$ επιπέδου). Η επιλογή αυτή έγινε διότι η γνώση της γενικής κατηγορίας των ερωτήσεων δεν βοηθάει τόσο στην απάντηση ερωτήσεων όσο εκείνη των ειδικών κατηγοριών.

Όπως προαναφέρθηκε, στα πειράματά μας χρησιμοποιήσαμε την υλοποίηση των ΜΔΥ που περιέχεται στο Weka, με τις προεπιλεγμένες τιμές των παραμέτρων της.[4] Το Weka είναι ένα εργαλείο μηχανικής μάθησης που περιέχει υλοποιήσεις των περισσοτέρων γνωστών αλγορίθμων μηχανικής μάθησης. Είναι γραμμένο σε JAVA και είναι σχετικά εύκολο στη χρήση του. Όσον αφορά τις ΜΔΥ, το Weka υλοποιεί τον αλγόριθμο Σειριακής Ελάχιστης Βελτιστοποίησης (Sequential Minimal Optimization - SMO) [28]. Αντίθετα, οι Zhang & Lee [39], χρησιμοποίησαν το LIBSVM [31], μια άλλη υλοποίηση ΜΔΥ, ενώ οι Li & Roth [18] χρησιμοποίησαν το SNoW [3], το οποίο ανήκει στην κατηγορία των αλγορίθμων που χρησιμοποιούν πυρήνα (kernel method), αλλά δεν είναι ΜΔΥ.

Στην απλούστερη περίπτωση που δοκιμάσαμε, κάθε ερώτηση παριστάνεται από ένα διάνυσμα του οποίου οι ιδιότητες έχουν τιμές 0 ή 1. Κάθε ιδιότητα αντιστοιχεί σε μια διαφορετική λέξη και δείχνει αν η λέξη εμφανίζεται στην ερώτηση ή όχι. Τα διανύσματα περιλαμβάνουν μία ιδιότητα για κάθε λέξη που εμφανίζεται πάνω από δύο φορές στη συλλογή εκπαίδευσης.

Αφού πετύχαμε αποτελέσματα παρόμοια με των προαναφερθέντων εργασιών, διερευνήσαμε το κατά πόσον τα αποτελέσματα βελτιώνονται χρησιμοποιώντας διαφορετικές διανυσματικές αναπαραστάσεις των ερωτήσεων. Πιο συγκεκριμένα, εξερευνήσαμε τα ακόλουθα:

α) Χρήση κατωφλιού για τη συχνότητα εμφάνισης των λέξεων. Απορρίπτουμε λέξεις που εμφανίζονται πολύ λίγες φορές, δηλαδή δεν τις αντιστοιχούμε σε ιδιότητες.

β) Επιλογή ιδιοτήτων με το μέτρο του πληροφοριακού κέρδους (information gain).

γ) Χρήση 2 ή 3 συνεχομένων λέξεων ως ιδιοτήτων (διγράμματα και τριγράμματα),

δ) Χρήση διγραμμάτων και τριγραμμάτων που δεν αποτελούνται απαραίτητα από συνεχόμενες λέξεις.

[4] Για τις ανάγκες της εργασίας χρησιμοποιήθηκε ένα σύστημα Pentium 4 στα 3,2 GHz με 1,5 GB RAM.

ε) Ειδική σήμανση της αρχής μιας ερώτησης και προσθήκη της ως λέξης.

στ) Απαλοιφή των λέξεων που αποτελούνται από τρία γράμματα και κάτω εκτός των ερωτηματικών μορίων. Έτσι απαλείφουμε συνδέσμους όπως το «and», «not», κ.τ.λ..

ζ) Χρήση ενός Εργαλείου Αναγνώρισης Κυρίων Ονομάτων (Named Entity Recognizer) για την αντικατάσταση των κυρίων ονομάτων στις ερωτήσεις με τους τύπους τους (πρόσωπο, τοποθεσία κλπ.).

η) Χρήση του WordNet, ώστε να προστίθενται σε κάθε ερώτηση τα συνώνυμα και υπερώνυμα του πρώτου ουσιαστικού της ερώτησης.

Παρακάτω θα αναλύσουμε μία προς μία τις ιδέες αυτές και τα αποτελέσματα που αυτές μας έδωσαν. Στο τέλος θα συγκρίνουμε όλα τα αποτελέσματα σε ένα διάγραμμα για να διαπιστωθούν οι διαφορές ανάμεσά τους.

4.2. ΧΡΗΣΗ ΚΑΤΩΦΛΙΟΥ

Επειδή μια γραμμική ΜΔΥ στηρίζεται στον υπολογισμό εσωτερικών γινομένων των διανυσμάτων εκπαίδευσης, η επεξεργαστική ισχύς που απαιτεί αυξάνεται με την αύξηση των ιδιοτήτων, με αποτέλεσμα να καθυστερεί η ολοκλήρωση των πειραμάτων. (Αυτό ήταν σημαντικό πρόβλημα στην περίπτωση της υλοποίησης που χρησιμοποιήσαμε, η οποία είναι ιδιαίτερα αργή με μεγάλα σύνολα ιδιοτήτων.) Επίσης, η χρήση πολύ μεγάλου αριθμού άχρηστων ιδιοτήτων ενδέχεται να προσθέτει θόρυβο στα δεδομένα εκπαίδευσης, εμποδίζοντας την ΜΔΥ να επιτύχει καλό διαχωρισμό των κατηγοριών. Έτσι είναι λογικό να προσπαθήσει κανείς να απαλείψει τις λιγότερο χρήσιμες ιδιότητες. Η πρώτη ιδέα ήταν να αφαιρέσουμε τις λέξεις (να μην τις αντιστοιχούμε σε ιδιότητες) που εμφανίζονται συνολικά κάτω από n φορές στο σύνολο εκπαίδευσης. Κάναμε πειράματα για διάφορες τιμές του n και βρήκαμε ότι η καλύτερη αντιστάθμιση του χρόνου εκτέλεσης των πειραμάτων με την επιθυμητή ακρίβεια επιτυγχάνεται στην περίπτωσή μας για n=3 (Οι Zhang & Lee [39] χρησιμοποίησαν κατώφλι ίσο με 10 στην εργασία τους.) Όλα αυτά γίνονται πιο κατανοητά αν κοιτάξουμε τον πίνακα παρακάτω:

Αριθμός ερωτήσεων εκπαίδευσης	Κατώφλι	Αριθμός λέξεων (ιδιοτήτων)	Επιτυχία
1000	1	2945	66,8%
2000	1	4892	73,6%
3000	1	6449	76%
4000	1	7832	77,8%
5452	1	9466	-
1000	3	398	64,4%
2000	3	833	73,0%
3000	3	1238	74,8%
4000	3	1653	75,8%
5452	3	2208	79,0%
1000	5	190	61,2%
2000	5	408	71,0%
3000	5	620	73,0%
4000	5	832	74,8%
5452	5	1177	78,8%

Για 5452 ερωτήσεις και κατώφλι 1 δεν κατορθώσαμε να ολοκληρώσουμε το πείραμα, διότι οι απαιτήσεων μνήμης της υλοποίησης ΜΔΥ που χρησιμοποιήσαμε ξεπερνούσαν τις δυνατότητες του υπολογιστή μας.

4.3. ΕΠΙΛΟΓΗ ΙΔΙΟΤΗΤΩΝ ΜΕ ΒΑΣΗ ΤΟ ΠΛΗΡΟΦΟΡΙΑΚΟ ΚΕΡΔΟΣ

Ένας άλλος τρόπος απαλοιφής ιδιοτήτων είναι η χρήση του μέτρου του πληροφοριακού κέρδους, ώστε να επιλεγούν μόνο οι ιδιότητες που δίνουν την περισσότερη πληροφορία στη ΜΔΥ. Χρησιμοποιήθηκε η σχετική βιβλιοθήκη «weka.attributeSelection.InfoGainAttributeEval». Οι μέθοδοι που παρέχει εκτιμούν την αξία μιας ιδιότητας Attribute μετρώντας το πληροφοριακό κέρδος που παρέχει σύμφωνα με τον παρακάτω τύπο:

$$InfoGain(Class,Attribute) = H(Class) - H(Class \mid Attribute)$$

όπου H(Class) η εντροπία της τυχαίας μεταβλητής Class που παριστάνει την κατηγορία της ερώτησης και H(Class|Attribute) η αναμενόμενη εντροπία της Class δοθείσης της τιμής της ιδιότητας Attribute.

Οι ιδιότητες στη συνέχεια κατατάσσονται με βάση το πληροφοριακό κέρδος που προσφέρουν (από το μεγαλύτερο πληροφοριακό κέρδος στο μικρότερο). Επιλέξαμε να κάνουμε πειράματα με το 20%, 40%, 60%, 80% των ιδιοτήτων, κρατώντας κάθε φορά εκείνες που προσφέρουν το μεγαλύτερο πληροφοριακό κέρδος. Τα καλύτερα αποτελέσματα τα είχαμε για το 60%. Τα πειράματα με βάση το πληροφοριακό κέρδος μείωσαν οριακά τα ποσοστά επιτυχίας για τα σύνολα ερωτήσεων των 1000, 2000, και 3000 ερωτήσεων μέχρι και 0,6% . Για τα σύνολα των 4000 και 5452 ερωτήσεων είχαμε μια οριακή αύξηση στα ποσοστά επιτυχίας κατά 0,4%. Τα παραπάνω γίνονται πιο εύληπτα με τον ακόλουθο πίνακα. Τα αποτελέσματά μας βρίσκονται εγγύτερα σε εκείνα των Zhang & Lee [39], που επίσης χρησιμοποίησαν μια ΜΔΥ.

Αριθμός ερωτήσεων εκπαίδευσης	Κατώφλι	Αριθμός λέξεων (προ Info Gain)	Info gain (ποσοστό λέξεων)	Τελικός αριθμός λέξεων (ιδιοτήτων)	Επιτυχία	Li-Roth (SNoW)	Zhang-Lee (LIBSVM)
1000	3	398	60%	239	64,2%	71,0%	68,0%
2000	3	833	60%	500	72,8%	77,8%	75,0%
3000	3	1238	60%	743	74,2%	79,8%	77,2%
4000	3	1653	60%	992	76,2%	80,0%	77,4%
5452	3	2208	60%	1325	79,4%	84,2%	80,2%

4.4. ΧΡΗΣΗ ΑΚΟΛΟΥΘΙΩΝ ΛΕΞΕΩΝ ΩΣ ΙΔΙΟΤΗΤΩΝ

Εκτός από μεμονωμένες λέξεις, μπορούμε να αντιστοιχίσουμε ιδιότητες και σε ακολουθίες δύο (διγράμματα) ή τριών (τριγράμματα) συνεχόμενων λέξεων, ελπίζοντας ότι υπάρχουν ακολουθίες λέξεων που σηματοδοτούν συγκεκριμένες κατηγορίες ερωτήσεων (π.χ. «How do I…», «Name the capital…»). Δυστυχώς τα αποτελέσματα έδειξαν πως με την προσθήκη ιδιοτήτων που αντιστοιχούν σε

διγράμματα τα ποσοστά επιτυχίας μειώθηκαν από 15% έως και 25% για κατώφλι συχνότητας εμφανίσεων 3 και 5, ενώ χωρίς κατώφλι (κατώφλι = 1) τα ποσοστά ήταν χαμηλότερα κατά 0,4% έως 1,6% από τα αρχικά. Αυτά τα αποτελέσματα φαίνεται να δείχνουν πως τα πιο συχνά διγράμματα (που παραμένουν με κατώφλι 3 και 5) συνεισφέρουν περισσότερο θόρυβο παρά χρήσιμη πληροφορία κι έτσι τα αποτελέσματα είναι χειρότερα από εκείνα που παίρνουμε με τη χρήση μόνο μεμονωμένων λέξεων. Όταν διατηρούνται και τα πιο σπάνια διγράμματα (κατώφλι = 1), τα αποτελέσματα βελτιώνονται αλλά και πάλι δεν υπερβαίνουν τα αποτελέσματα που είχαμε χρησιμοποιώντας μεμονωμένες μόνο λέξεις. Με την προσθήκη και τριγραμμάτων (και κατώφλι = 1), τα αποτελέσματα ήταν τα ίδια ακριβώς με εκείνα που πήραμε χρησιμοποιώντας ως ιδιότητες μεμονωμένες λέξεις. Δεν μελετήθηκε η περίπτωση αξιολόγησης των διγραμμάτων και τριγραμμάτων με το μέτρο του πληροφοριακού κέρδους.

Συνοψίζοντας, η προσθήκη ιδιοτήτων που αντιστοιχούν σε διγράμματα και τριγράμματα δεν έδωσε καλύτερα αποτελέσματα. Παράλληλα, τα δεδομένα εκπαίδευσης αυξήθηκαν σε μέγεθος – έως και 2 φορές μεγαλύτερα – με αποτέλεσμα η χρήση τους στο Weka να γίνεται δυσχερής (για τα μικρά σύνολα ερωτήσεων εκπαίδευσης) έως αδύνατη (για τα μεγάλα σύνολα – 4000 και 5452 ερωτήσεων). Παρακάτω παρατίθενται τα σχετικά αποτελέσματα. Για το σύνολο των 5452 ερωτήσεων εκπαίδευσης, καθώς και για την περίπτωση των τριγραμμάτων και το σύνολο των 4000 ερωτήσεων εκπαίδευσης, δεν υπάρχουν αποτελέσματα. Αυτό οφείλεται και πάλι στην αδυναμία ολοκλήρωσης των σχετικών πειραμάτων λόγω των απαιτήσεων σε μνήμη της υλοποίησης ΜΔΥ που χρησιμοποιήσαμε.

Αριθμός ερωτήσεων εκπαίδευσης	Κατώφλι	Επιτυχία (μεμονωμένες λέξεις)	Επιτυχία (λέξεις + διγράμματα)	Επιτυχία (λέξεις + διγράμματα + τριγράμματα)
1000	1	66,8%	66,4%	66,4%
2000	1	73,6%	71,2%	71,2%
3000	1	76,0%	74,0%	74,0%
4000	1	77,8%	76,2%	-
5452	1	-	-	-

4.5. ΧΡΗΣΗ ΜΗ ΣΥΝΕΧΟΜΕΝΩΝ ΔΙΓΡΑΜΜΑΤΩΝ ΚΑΙ ΤΡΙΓΡΑΜΜΑΤΩΝ

Αυτή η ιδέα αποτελεί τη συνέχεια της προηγούμενης με την μόνη αλλαγή ότι επιτρέπεται να παραληφθούν μία ή δύο λέξεις ανάμεσα από δύο συνεχόμενες λέξεις που συμμετέχουν σε ένα δίγραμμα ή τρίγραμμα. Επί παραδείγματι για την ερώτηση «Who was President of Cuba in 1978 ?», οι ιδιότητες που θα παίρναμε για διγράμματα με παράλειψη μίας λέξης, θα ήταν οι εξής: «Who President», «was of», «President Cuba», «of in», «Cuba 1978», «in ?». Η σημαντικότερη παρατήρηση για τα αποτελέσματα με παραλήψεις λέξεων είναι πως διατηρούν τα ίδια αποτελέσματα με τα αντίστοιχα των απλών διγραμμάτων και τριγραμμάτων. Αυτό φαίνεται αναλυτικότερα στον παρακάτω πίνακα:

Αριθμός ερωτήσεων	Κατώφλι	Επιτυχία (μεμονωμένες λέξεις)	Επιτυχία (λέξεις + διγράμματα με παραλήψεις ως 1 λέξης)	Επιτυχία (λέξεις + διγράμματα με παραλήψεις ως 2 λέξεων)	Επιτυχία (λέξεις + διγράμματα + τριγράμματα με παραλήψεις ως 1 λέξης)	Επιτυχία (λέξεις + διγράμματα + τριγράμματα με παραλήψεις ως 2 λέξεων)
1000	1	66,8%	66,4%	66,4%	66,4%	66,4%
2000	1	73,6%	71,2%	71,2%	71,2%	71,2%
3000	1	76,0%	74,0%	74,0%	74,0%	74,0%
4000	1	77,8%	76,2%	76,2%	-	-
5452	1	-	-	-	-	-

Όπου υπάρχουν κενά στον παραπάνω πίνακα, δεν καταφέραμε να παραγάγουμε αποτελέσματα, λόγω των απαιτήσεων σε μνήμη της υλοποίησης ΜΔΥ που χρησιμοποιήσαμε.

4.6. ΧΡΗΣΗ ΕΝΟΣ ΣΥΣΤΗΜΑΤΟΣ ΑΝΑΓΝΩΡΙΣΗΣ ΚΥΡΙΩΝ ΟΝΟΜΑΤΩΝ

Ένα Σύστημα Αναγνώρισης Κυρίων Ονομάτων (ΣΑΚΟ, Named Entity Recognizer – NER) έχει τη δυνατότητα να αναγνωρίσει τα κύρια ονόματα μέσα σε ένα κείμενο, καθώς και τους τύπους των κυρίων ονομάτων (πρόσωπο, τοποθεσία κλπ). Όπως προαναφέρθηκε, χρησιμοποιήσαμε το ΣΑΚΟ που περιλαμβάνεται στο σύστημα επεξεργασίας φυσικής γλώσσας GATE (έκδοση 3.0beta).

Το ΣΑΚΟ του GATE υποστηρίζει πολλούς τύπους κυρίων ονομάτων. Εμείς επιλέξαμε να χρησιμοποιήσουμε τους παρακάτω: «Organization», «Location», «Person», «Title» και «Date». Κάθε κύριο όνομα που ανακαλύφθηκε στις ερωτήσεις (εκπαίδευσης και αξιολόγησης) αντικαταστάθηκε από τον τύπο του, ο οποίος εισήχθη στην ερώτηση ως λέξη («organization», «location» κλπ), προκειμένου να δοθεί στη ΜΔΥ η δυνατότητα μεγαλύτερης γενίκευσης (π.χ. οι ερωτήσεις «What is the capital of Greece?» και «What is the capital of Italy?» παριστάνονται πλέον από το ίδιο διάνυσμα). Στη συνέχεια αντιστοιχίσαμε τις λέξεις των ερωτήσεων εκπαίδευσης σε ιδιότητες, όπως στην ενότητα 4.2. Τα αποτελέσματα ήταν οριακά καλύτερα μόνο για τις 2000 ερωτήσεις εκπαίδευσης, όπου το ποσοστό επιτυχίας αυξήθηκε κατά 0,4%. Στην περίπτωση των υπολοίπων συνόλων ερωτήσεων εκπαίδευσης, τα ποσοστά είτε παρέμειναν σταθερά είτε μειώθηκαν ελαφρά. Παρακάτω φαίνονται τα αποτελέσματα αναλυτικά. Δεν διερευνήθηκε η χρήση ΣΑΚΟ σε συνδυασμό με διγράμματα και τριγράμματα.

Αριθμός ερωτήσεων εκπαίδευσης	Κατώφλι	Επιτυχία	Επιτυχία με χρήση ΣΑΚΟ
1000	1	66,8%	66,8%
2000	1	73,6%	74,0%
3000	1	76,0%	75,8%
4000	1	77,8%	77,4%
5452	1	-	-

Τα κενά στον παραπάνω πίνακα οφείλονται στις απαιτήσεις σε μνήμη της υλοποίησης ΜΔΥ που χρησιμοποιήσαμε.

4.7. ΧΡΗΣΗ ΤΟΥ WORDNET

Στις περισσότερες ερωτήσεις υπάρχει ένα «κεντρικό» ουσιαστικό που παρέχει σημαντικές πληροφορίες για τον τύπο της απάντησης που ζητά η ερώτηση (π.χ. «Ποια είναι η πρωτεύουσα της Ελλάδας;», «Ποιος είναι ο πρόεδρος της Αργεντινής;»). Ο εντοπισμός του κεντρικού ουσιαστικού μπορεί να παίξει σημαντικό ρόλο στην εύρεση της κατηγορίας της ερώτησης, όπως έχουν δείξει προηγούμενες εργασίες [10] [8]. Στην εργασία [29] αναφέρεται, ακόμη, ότι η προσθήκη στις ερωτήσεις εκπαίδευσης επιπλέον λέξεων που σχετίζονται με το κεντρικό ουσιαστικό τους (π.χ. συνώνυμα, υπερώνυμα) βοηθά πολύ στην επιτυχή κατάταξη καινούριων ερωτήσεων. Στις εργασίες [20] [26] [23] χρησιμοποιείται, επίσης, το WordNet για την προσθήκη υπερωνύμων κατά την κατάταξη ερωτήσεων ή γενικότερα κειμένων.

Βασιζόμενοι στις ιδέες των προσεγγίσεων της προηγούμενης παραγράφου, μπορούμε να παρατηρήσουμε ότι δύο ερωτήσεις όπως «What is the largest country in Europe?» και «What is the most populated region in Germany?» περιέχουν κεντρικά ουσιαστικά («country» και «region») που σχετίζονται σημασιολογικά (και τα δύο αναφέρονται σε γεωγραφικές τοποθεσίες). Η συνάφεια μεταξύ των δύο ερωτήσεων μπορεί να γίνει περισσότερο εμφανής αν αντικατασταθούν τα δύο κεντρικά ουσιαστικά με μια λέξη (π.χ. «location») που αντιπροσωπεύει τον κοινό σημασιολογικό τους τύπο, με τον ίδιο τρόπο που η χρήση ενός ΣΑΚΟ αντικαθιστά κύρια ονόματα με τους τύπους τους. Γενικότερα, προσθέτοντας σε κάθε ερώτηση συνώνυμα ή υπερώνυμα (λέξεις με ευρύτερη σημασία) του κεντρικού ουσιαστικού της, αυξάνεται η πιθανότητα ερωτήσεις που ζητούν συναφείς σημασιολογικά απαντήσεις να περιέχουν κοινές λέξεις και άρα να οδηγούν σε κοινές τιμές ιδιοτήτων στα διανύσματα που τις παριστάνουν. Αυτό είναι δυνατόν να βοηθήσει τον αλγόριθμο μηχανικής μάθησης να κατατάξει στην ίδια κατηγορία ερωτήσεις που ζητούν συναφείς σημασιολογικά απαντήσεις.

Στην παρούσα εργασία, η προσθήκη συνωνύμων και υπερωνύμων επιτεύχθηκε με τη βοήθεια του WordNet. Το WordNet είναι ένας ιεραρχικός θησαυρός λέξεων που παρέχει, μεταξύ άλλων, τις εξής πληροφορίες για κάθε λέξη λ που περιέχει:

- Μέρος του λόγου (ρήμα, ουσιαστικό κλπ).
- Συνώνυμα (λέξεις που βρίσκονται στο ίδιο επίπεδο με τη λ),
- Υπερώνυμα (λέξεις που βρίσκονται πιο ψηλά στην ιεραρχία – παριστάνουν πιο γενικές έννοιες από ό,τι η λ),
- Υπώνυμα (λέξεις που βρίσκονται πιο χαμηλά στην ιεραρχία – πιο ειδικές έννοιες).

Επειδή δεν υπήρχε εργαλείο ελεύθερα διαθέσιμο για την εύρεση του κεντρικού ουσιαστικού κάθε ερώτησης, ούτε και ο απαραίτητος χρόνος για να αναπτυχθεί εξαρχής ένα τέτοιο εργαλείο, αποφασίσαμε να χρησιμοποιήσουμε ως κεντρικό ουσιαστικό την πρώτη λέξη της ερώτησης που αναγνωρίζει ως ουσιαστικό το WordNet.[5] Σε κάθε ερώτηση προσθέταμε τα υπερώνυμα και τα συνώνυμα αυτής της λέξης. Η επιλογή αυτή αποδείχθηκε πως δίνει καλά αποτελέσματα, καθώς οι ερωτήσεις έχουν κατά μέσο όρο 1-2 ουσιαστικά και το πρώτο που συναντούμε είναι, στην πλειοψηφία των περιπτώσεων, το κεντρικό. Μετά από πειράματα, αποφασίσαμε

[5] Το WordNet παρέχει συναρτήσεις που μπορούν να χρησιμοποιηθούν για αυτόν το σκοπό. Παράλληλα, το WordNet μπορεί να αναγνωρίσει εάν ένα ουσιαστικό βρίσκεται σε πληθυντικό αριθμό και να το μετατρέψει στη βασική του μορφή (ενικό αριθμό).

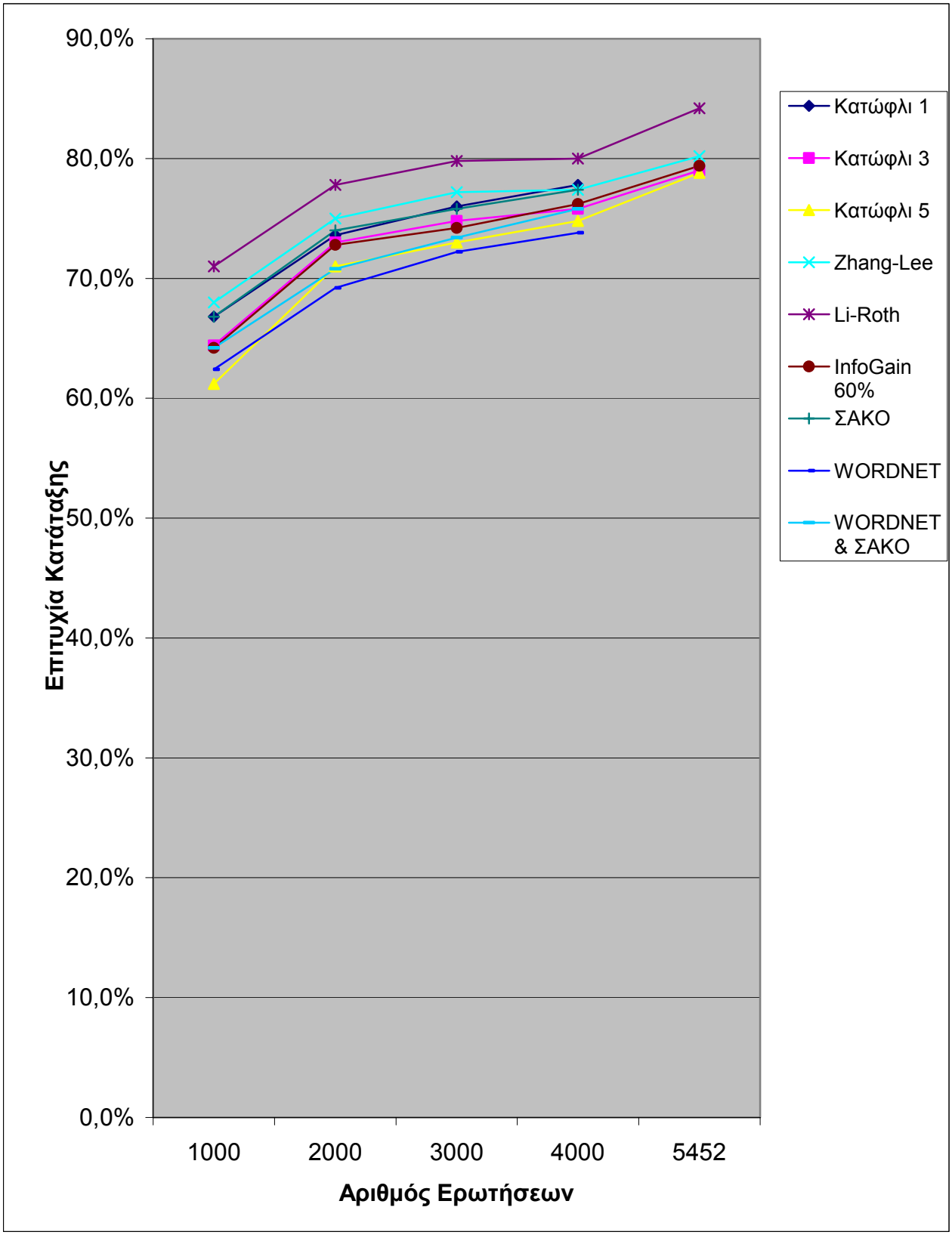
να προσθέτουμε όλα τα υπερώνυμα του κεντρικού ουσιαστικού (αντί π.χ. των υπερωνύμων συγκεκριμένου βάθους). Τα ποσοστά που επετεύχθησαν από την προσθήκη των υπερωνύμων φαίνονται παρακάτω. Δυστυχώς, ούτε και με την προσθήκη των υπερωνύμων υπήρξε βελτίωση των αποτελεσμάτων, όπως φαίνεται και στον πίνακα.

Αριθμός ερωτήσεων εκπαίδευσης	Κατώφλι	Επιτυχία	Επιτυχία με χρήση WORDNET	Επιτυχία με χρήση WORNDET & ΣΑΚΟ
1000	1	66,8%	62,4%	64,2%
2000	1	73,6%	69,2%	70,8%
3000	1	76,0%	72,2%	73,4%
4000	1	77,8%	73,8%	75,8%
5452	1	-	-	-

Σημειώνουμε και πάλι πως το μεγάλο πλήθος των ιδιοτήτων δεν επέτρεψε την ολοκλήρωση των πειραμάτων για το σύνολο εκπαίδευσης των 5452 ερωτήσεων (μεγάλες απαιτήσεις σε μνήμη της υλοποίησης ΜΔΥ που χρησιμοποιήσαμε).

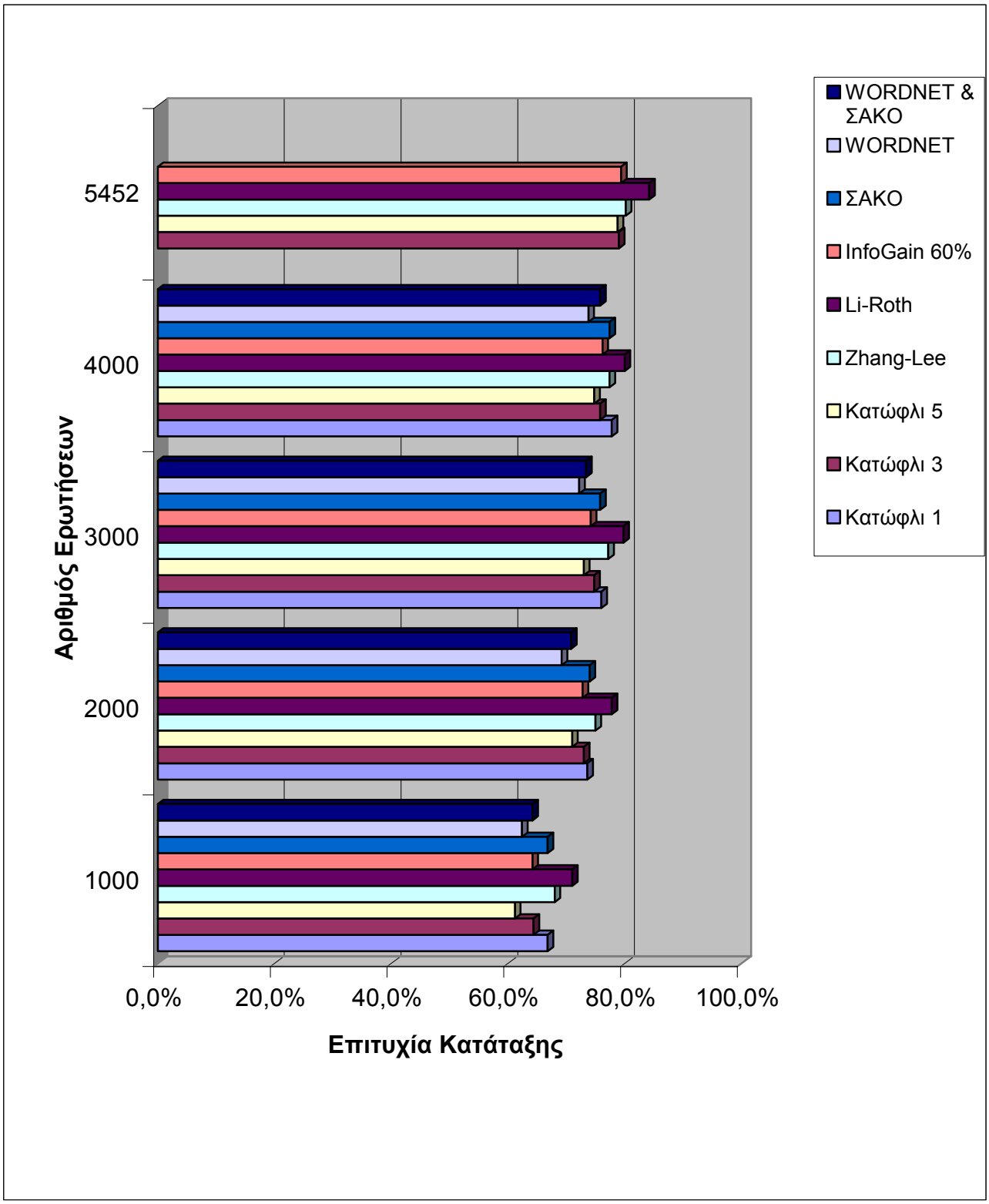
4.8 ΣΥΓΚΕΝΤΡΩΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ

Παρακάτω φαίνονται τα αποτελέσματα των πειραμάτων συγκεντρωμένα σε διαγράμματα. Γίνεται φανερό πως τα αποτελέσματά μας υπολείπονται από αυτά των Zhang & Lee [39] και Li & Roth [18], αν και βρίσκονται σχετικά κοντά στα αποτελέσματα των Zhang & Lee, που χρησιμοποίησαν επίσης μια υλοποίηση ΜΔΥ. Στην περίπτωση της χρήσης μόνο μεμονωμένων λέξεων, οι διαφορές των αποτελεσμάτων μας από εκείνων των Zhang & Lee [39] οφείλονται κατά πάσα πιθανότητα στη χρήση διαφορετικής υλοποίησης ΜΔΥ (χρησιμοποίησαν το LIBSVM αντί για το Weka), ενδεχομένως στις διαφορετικές τιμές των παραμέτρων των υλοποιήσεων (χρησιμοποιήσαμε τις προεπιλεγμένες τιμές του Weka) και στη χρήση διαφορετικού κατωφλίου (οι Zhang & Lee χρησιμοποίησαν κατώφλι 10). Ακόμα και στην περίπτωση που χρησιμοποιήσαμε ΣΑΚΟ ή τα υπερώνυμα του WordNet – και ενώ περιμέναμε τα αποτελέσματα να είναι καλύτερα – δεν είχαμε βελτίωση. Μεταξύ των παραλλαγών που δοκιμάσαμε, καλύτερα αποτελέσματα είχαμε χρησιμοποιώντας μόνο ιδιότητες που αντιστοιχούν σε μεμονωμένες λέξεις, χωρίς επιλογή ιδιοτήτων με κατώφλι συχνότητας ή πληροφοριακό κέρδος. Τα αποτελέσματα πάντως σε όλες τις περιπτώσεις δεν δείχνουν συμπτώματα κορεσμού, κάτι που μας κάνει να πιστεύουμε πως είναι δυνατόν να επιτευχθούν καλύτερα αποτελέσματα με μεγαλύτερες συλλογές ερωτήσεων εκπαίδευσης, κάτι που απαιτεί και ταχύτερη υλοποίηση ΜΔΥ από αυτή που χρησιμοποιήσαμε.



Διάγραμμα 1. Πειραματικά αποτελέσματα.

Το Διάγραμμα 2 μας δίνει πιο καθαρά τα αποτελέσματα του διαγράμματος 1.



Διάγραμμα 2. Τα πειραματικά αποτελέσματα σε μορφή ραβδοδιαγράμματος.

5. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΙΘΑΝΕΣ ΒΕΛΤΙΩΣΕΙΣ

Καταφέραμε να επιβεβαιώσουμε τα πειραματικά αποτελέσματα των Zhang & Lee [39], με μικρές διαφορές που κατά πάσα πιθανότητα οφείλονται στις διαφορετικές υλοποιήσεις ΜΔΥ που χρησιμοποιήσαμε, στις διαφορετικές τιμές των παραμέτρων των υλοποιήσεων και στο διαφορετικό κατώφλι συχνότητας λέξεων. Τα αποτελέσματά μας ήταν κατώτερα από εκείνα των Li & Roth [18], που χρησιμοποίησαν διαφορετικό αλγόριθμο μάθησης (SNoW [3]). Επιχειρήσαμε να βελτιώσουμε περαιτέρω τα αποτελέσματά μας χρησιμοποιώντας επιλογή ιδιοτήτων, ιδιότητες που αντιστοιχούν σε n-γράμματα, αντικατάσταση κυρίων ονομάτων από τους τύπους τους και επέκταση των ερωτήσεων με υπερώνυμα και συνώνυμα του κεντρικού ουσιαστικού, χωρίς όμως να επιτύχουμε βελτίωση.

Τα πειράματα έγιναν με χρήση πολυωνυμικού πυρήνα βαθμού 1, κάτι που πρακτικά σημαίνει πως η ΜΔΥ επιχειρεί να κατασκευάσει το υπερεπίπεδο μεγίστου περιθωρίου στον αρχικό διανυσματικό χώρο, κάτι που είναι απολύτως εφικτό μόνο όταν τα δεδομένα εκπαίδευσης είναι γραμμικά διαχωρίσιμα. Θα θέλαμε να έχουμε την ευχέρεια να δοκιμάσουμε πυρήνα με πολυωνυμικό βαθμό 2 ή και παραπάνω αλλά αυτό δεν ήταν δυνατόν λόγω της μικρής ταχύτητας και των μεγάλων απαιτήσεων μνήμης της υλοποίησης ΜΔΥ που χρησιμοποιήσαμε. Θα ήταν επίσης χρήσιμο να δοκιμάσουμε και άλλους πυρήνες, όπως για παράδειγμα ο RBF, αλλά και πάλι αυτό δεν ήταν δυνατόν, επειδή δεν τους υποστήριζε η υλοποίηση ΜΔΥ που χρησιμοποιήσαμε.

Η εύρεση του κεντρικού ουσιαστικού έγινε προσεγγιστικά, εντοπίζοντας το πρώτο ουσιαστικό της ερώτησης. Η προσθήκη υπερωνύμων ενδέχεται να έδινε καλύτερα αποτελέσματα αν ο εντοπισμός του κεντρικού ουσιαστικού γινόταν με μεγαλύτερη ακρίβεια. Η προσθήκη ιδιοτήτων που έχουν σχέση με το συντακτικό της ερώτησης είναι μια ακόμα δυνατή βελτίωση που έχει αποδειχθεί ότι βοηθάει [17][12].

ΑΝΑΦΟΡΕΣ

[1] *Banko M., Brill E. , "Scaling to very very large corpora for natural language disambiguation", In Meeting of the Association for Computational Linguistics (2001), pages 26–33.*

[2] *Burges C.J.C., "A Tutorial on Support Vector Machines for Pattern Recognition", in Data Mining and Knowledge Discovery 2, 121-167, 1998.*

[3] *Carlson C., Cumby J., Rosen and Roth D.. "SNoW User Guide. Technical Report" UIUCDCS-R-99-2101. UIUC Computer Science Department, Aug 1999. http://l2r.cs.uiuc.edu/~danr/snow.html.*

[4] *Chang C-C and Lin C-J. "LIBSVM: a library for support vector machines". 2001. http://www.csie.ntu.edu.tw/~cjlin/libsvm*

[5] *Cristianini N., Shawe-Taylor J., "An Introduction to Support Vector Machines and Other Kernel-based Learning Methods" , Cambridge Univ. Press, 2000.*

[6] *Diekema R., Yilmazel O., Liddy E. D., "Evaluation of Restricted-Domain Question-Answering Systems", ACL 2004.*

[7] *Fellbaum Ch., Wordnet, An Electronic Lexical Database, MIT Press, Cambridge Massachusetts, 1999*

[8] *Ferret O., Grau B., Hurault-Plantet M., Illouz G., Monceaux L., Robba I., Vilnat A., "Finding an answer based on the recognition of the question focus", TREC 2001.*

[9] *Gunn S., "Support Vector Machines for Classification and Regression", Image Speech & Intelligent Systems Group, University of Southampton, May 1998.*

[10] *Harabagiu S. M., Maiorano S. J., Pasca M. A., "Open-domain textual question answering techniques", Natural Language Engineering 9 (3): 2321-267, 2003 Cambridge University Press.*

[11] *Harabagiu S., Moldovan D., Pasca M., Mihalcea R., Surdeanu M., Bunescu R., Girju R., Rus V., Morarescu P., "The Role of Lexico-Semantic Feedback in Open-Domain Textual Question-Answering", Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL-2001), July 2001, Toulouse, France, 274-281.*

[12] *Hermjakob U., "Parsing and Question Classification for Question Answering", Proceedings of the Workshop on Open-Domain Question Answering at ACL-2001, Toulouse, France, July 6-11, 2001.*

[13] *Hovy E., Gerber L., Hermjakob U., Lin C. and Ravichandran D.. 2001. "Toward semantics-based answer pinpointing", In Proceedings of the Darpa Human Language Technology Conference (HLT). San Diego, CA, 2001.*

[14] *Hsu C.-W. and Lin C.-J., "A Comparison of Methods for Multi-class Support Vector Machines", IEEE Trans. Neural Networks, vol. 13, no. 2, pp. 415-425, March 2001.*

[15] *Hsu C-W, Chang C-C, Lin C-J, "A Practical Guide to Support Vector Classification", http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html.*

[16] *Joachims T., "Support Vector Machines", KI 13(4): 54-55 (1999).*

[17] *Li W., "Question Classification Using Language modelling", in CIIR Technical Report: University of Massachusetts, Amherst, 2002..*

[18] *Li X., Roth D., "Learning Question Classifiers", in Proceedings of the 19th International Conference on Computational Linguistics (COLING), Taipei, Taiwan, 2002, pp. 556-562.*

[19] *McCallum and Nigam K.. A Comparison of Event Models for Naïve Bayes Text Classification. In AAAI- 98 Workshop on Learning for Text Categorization, 1998.*

[20] *McCallum R., Rosenfeld T., Mitchell A., Ng Y., "Improving Text Classification by Shrinkage in a Hierarchy of Classes", Proceedings of (ICML)-98, 15th International Conference on Machine Learning, pp. 359-367, 1998.*

[21] *Miliaraki S., Androutsopoulos I., "Learning to Identify Single-Snippet Answers to Definition Questions", Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004), Geneve, Switzerland, pp. 1360*

[22] *Mitchell T.M., "Machine Learning", McGraw-Hill International Editions, 1997*

[23] *Na S-H, Kang I-S, Lee S-Y, Lee J-H, "Question Answering Approach Using a WordNet-based Answer Type Taxonomy", TREC 2002.*

[24] *Osouna E., Freund R., Girosi F., "Training Support Vector Machines: an Application to Face Detection", Center for Biological and Computational Learning and Operations Research Center, Massachusetts Institute of Technology, Proceedings of CVPR'97, Puerto Rico, 1997.*

[25] *Pasca M. A., Harabagiu S. M., "High Performance Question/Answering", SIGIR'01.*

[26] *Pinto D., Branstein M., Coleman R., Croft W. B., King M., Li W., Wei X., "QuASM: A System for Question Answering Using Semi-Structured Data", Proceedings of the JCDL 2002 Joint Conference on Digital Libraries, 2002.*

[27] *Plamondon L., Lapalme G., Kosseim L., "The QUANTUM Question Answering System", in Proceedings of the Tenth Text Retrieval Conference (TREC-X, p. 157-165, Gaithersburg, Maryland, 2001.*

[28] *Platt J., "Fast Training of Support Vector Machines using Sequential Minimal Optimization". Advances in Kernel Methods - Support Vector Learning, B. Sch□lkopf, C. Burges, and A. Smola, eds., MIT Press 1998.*

[29] *Prager J., Chu-Carroll J., Czuba K., "Use of WordNet Hypernyms for Answering What-is Questions", TREC 2001.*

[30] *Qi H., Otterbacher J., Winkel A., Radev D. R., "The University of Michigan at TREC2002: Question Answering and Novelty tracks".*

[31] *Quinlan J. R.. C4.5: Programs for Machine Learning. Morgan Kaufmann, San Francisco, CA, 1993.*

[32] *Radev D., Fan W., Qi H., Wu H., Grewal A., "Probabilistic Question Answering on the Web", Journal of the American Society for Information Science and Technology 56(3), March 2005.*

[33] *Ravichandran D., Hovy E., "Learning Surface Text Patterns", in Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia, July 2002, pp. 41-47.*

[34] *Srihari R., Li W., "A Question Answering System Supported by Information Extraction", ANLP 2000: pp. 166-172.*

[35] *Solorio1 T., Perez-Coutino1 M., Montes-y-Gomez M., Villasenor-Pineda1 L., Lopez-Lopez A., "A Language Independent Method for Question Classification", COLING-04, Geneva, Switzerland, 2004*

[36] *Suzuki J., Taira H., Sasaki Y., and Maeda E., "Question Classification using HDAG Kernel", Workshop on Multilingual Summarization and Question Answering, ACL 2003, Sapporo, Japan, pp.61-68*

[37] *Voorhees H., The TREC question answering track, Natural Language Engineering 7 (4): pp. 361-378, 2001 Cambridge University Press.*

[38] *Yang Y. and Liu X.. A Re-examination of Text Categorization Methods. In Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99), pp. 42-49, 1999.*

[39] *Zhang D., Lee W. S., "Question Classification using Support Vector Machines", SIGIR 2003, pp. 26-32.*