ΑΛΕΞΑΝΔΡΕΙΟ ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ
ΙΔΡΥΜΑ ΘΕΣΣΑΛΟΝΙΚΗΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΕΥΦΥΕΙΣ ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ – WEB INTELLIGENCE

Μηχανές αναζήτησης με εκτεταμένη χρήση ταξινομιών

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Θεόδωρου Άξιου

Επιβλέπων : Μιχάλης Σαλαμπάσης
Καθηγητής, ΑΤΕΙΘ

Θεσσαλονίκη, Σεπτέμβριος 2018

ΑΛΕΞΑΝΔΡΕΙΟ ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΘΕΣΣΑΛΟΝΙΚΗΣ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΕΥΦΥΕΙΣ ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ - WEBINTELLIGENCE

Μηχανές αναζήτησης με εκτεταμένη χρήση ταξινομιών

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Θεόδωρου Άξιου

Επιβλέπων : Μιχαήλ Σαλαμπάσης
Καθηγητής Α.Τ.Ε.Ι.Θ.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή στις 18 Σεπτεμβρίου 2018.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Μιχαήλ Σαλαμπάσης	Κωνσταντίνος Διαμαντάρας	Κέρστιν Σιάκα
Καθηγητής Α.Τ.Ε.Ι.Θ.	Καθηγητής Α.Τ.Ε.Ι.Θ.	Καθηγητής Α.Τ.Ε.Ι.Θ.

Θεσσαλονίκη, Σεπτέμβριος 2018

(Υπογραφή)

.....

Άξιος Θεόδωρος

Μηχανικός Πληροφορικής Α.Τ.Ε.Ι.Θ.

Ευχαριστίες

Στο σημείο αυτό θα ήθελα να ευχαριστήσω θερμά τον καθηγητή μου κ. Μιχάλη Σαλαμπάση για την εμπιστοσύνη που μου έδειξε κατά τη διάρκεια υλοποίησης της πτυχιακής μου εργασίας, όπως επίσης και για την πολύτιμη βοήθεια και καθοδήγηση του.

Θα ήθελα να ευχαριστήσω θερμά τον Μητροφάνη Ακριτόπουλο χωρίς την βοήθεια του οποίου δεν θα είχα την δυνατότητα να παρακολουθήσω το μεταπτυχιακό πρόγραμμα σπουδών.

Θα ήθελα επίσης να απευθύνω τις ευχαριστίες μου στους φίλους και συναδέλφους μου Ανδρέα Κουτούλα, Γιώργο Γεωργιάδη, Σοφοκλή Κέντρα, οι οποίοι με την συμπαράσταση τους συνέβαλαν στην εκπλήρωση του στόχου μου.

Περίληψη

Η παρούσα διπλωματική εργασία εξετάζει την υλοποίηση διαδικτυακής μηχανής αναζήτησης η οποία δέχεται ερωτήματα που δίνονται σε ‘φυσική γλώσσα’ τα οποία επεξεργάζεται βασιζόμενη σε τεχνικές NLP (Natural Language Processing) και στην συνέχεια προσπαθεί να απαντήσει στο ερώτημα του χρήστη.

Αναπτύχθηκε ένας μηχανισμός όπου παίρνει ως είσοδο, κείμενα φυσικής γλώσσας και χρησιμοποιώντας το Google NLP API αναλύει συντακτικά, σημασιολογικά, μορφολογικά, συναισθηματικά και γραμματικά, συναισθηματικά τα κείμενα. Στην συνέχεια εξάγει μεταδεδομένα για τα κείμενα αυτά Part of Speech POS, sentiment magnitude, αναγνωρίζει και εξάγει τις οντότητες που εμπεριέχονται σε αυτά. Κατόπιν κάνει χρήση της ανάλυσης και των μεταδεδομένων αυτών αντιστοιχίζει το ερώτημα σε κάποιο προκαθορισμένο πρότυπο και αποστέλλει ερώτημα στη μηχανή αναζήτησης. Η μηχανή αναζήτησης που είναι υλοποιημένη σε Elasticsearch αναζητά στον index της, ο οποίος βασίζεται σε δεδομένα που αφορούν ταινίες και ηθοποιούς με βάση του dataset της IMDB, και παρουσιάζει τα αποτελέσματα μέσω διαδικτυακής διεπαφής. Σε αυτή την φάση γίνεται χρήση τεχνικών αναζήτησης σε προκαθορισμένες οντότητες, faceted search, fielded search, spelling correction.

Λέξεις Κλειδιά: Επεξεργασία φυσικής γλώσσας, μηχανές αναζήτησης, ανάκτηση πληροφοριών, εξαγωγή οντοτήτων, Google NLP API

Abstract

This diploma thesis examines the implementation of the web-based search engine that receives text questions in "natural language", process them based on NLP techniques (Natural Language Processing) and then attempts to answer the question of the user.

A mechanism was developed where it takes as input, in natural language texts and uses the Google NLP API to analyze the text syntactically, semantically, morphologically, emotionally and grammatically, sentimentally. It then extracts metadata for parts of Speech POS, sentiment magnitude, recognizes and extracts the entities contained in them. Then it uses the analysis and metadata generated to match the query to a predefined template and sends a query request to the search engine. The search engine that is implemented in Elasticsearch, searches its index, consisting data on movies and actors based on the IMDB dataset, and presents the results through a web interface. In this phase, the following search techniques are applied to predefined taxonomies, faceted search, fielded search, spelling correction

Keywords: Natural Language Processing, search engine, information retrieval, entity extraction, Google NLP API

Πίνακας περιεχομένων

Πίνακας εικόνων

1

Εισαγωγή

1.1 Μηχανές αναζήτησης με εκτεταμένη χρήση ταξινομιών

Στον τομέα των υπολογιστών έχουν γίνει το τελευταίο διάστημα σπουδαία βήματα στην κατανόησης της γλώσσας. Έχουν αναπτυχθεί προγράμματα για επεξεργασία φυσικής γλώσσας που δίνουν τη δυνατότητα προσδιορισμού όλων των χαρακτηριστικών για κάθε λέξη μέσα σε κάθε πρόταση (γλωσσολογικά, συντακτικά, σημασιολογικά, συναίσθημα). Τα σημασιολογικά χαρακτηριστικά κάθε λέξης είναι πολύ πιο σημαντικά από ό,τι οι γλωσσολόγοι γενικά ήταν πρόθυμοι να αναγνωρίσουν. Υπάρχει ακόμη μια αυξανόμενη αναγνώριση του γεγονότος ότι το γενικό πλαίσιο στο οποίο εξετάζεται η γλώσσα είναι μέγιστης σημασίας κατά την ερμηνεία ενός κειμένου καθώς η έμμεση γνώση για τον πραγματικό κόσμο είναι αυτή που εφαρμόζεται συχνότερα και μπορεί να είναι πολύ καλά δομημένη ώστε να υποστηρίζει απόλυτα την κατανόηση και επεξεργασία της γλώσσας. Παρατηρούμε ακόμη ότι οι μηχανές αναζήτησης που κάνουν χρήση επεξεργασίας φυσικής γλώσσας γίνονται ευρέος διαδεδομένες, καθώς επιτρέπουν μια διεπαφή όπου ο χρήστης χρειάζεται να γνωρίζει λιγότερα τεχνικά στοιχεία για την επιμέρους εφαρμογή, κάνοντας χρήση πεδίων ελευθέρου κειμένου και αναγνώρισης φωνής οι μηχανές αναζήτησης βελτιώνουν κατακόρυφα την ευχρηστία τους, γίνονται καθ΄ αυτό τον τρόπο ελκυστικότερες στους λιγότερο τεχνικά καταρτισμένους χρήστες.

Έχοντας ως βάση το παραπάνω πλαίσιο και με δεδομένο γενικότερα ότι η επιστήμη της πληροφορικής έχει εξελιχθεί σε πάρα πολύ μεγάλο βαθμό τα τελευταία χρόνια η χρήση όλο και περισσότερων εφαρμογών στον τομέα της επεξεργασίας φυσικής γλώσσας γνωρίζει ιδιαίτερη

ανάπτυξη. Έτσι, μέσα κι από αυτό την οπτική της επεξεργασία φυσικής γλώσσας η επιστήμη της πληροφορικής παίζει ίσως το σπουδαιότερο ρόλο στην καθημερινότητά μας σε όλους τους τομείς όπως στην υγεία, στην ψυχαγωγία, στον πολιτισμό, την οικονομία, την ασφάλεια και την εκπαίδευση.

1.2 Αντικείμενο διπλωματικής

Στόχος της πτυχιακής είναι η υλοποίηση διαδικτυακής μηχανής αναζήτησης η οποία δέχεται ερωτήματα που δίνονται σε 'φυσική γλώσσα' από τον χρήστη, βασιζόμενη σε τεχνικές NLP (Natural Language Processing), επεξεργάζεται συντακτικά, σημασιολογικά, μορφολογικά και γραμματικά τα ερωτήματα και στην συνέχεια προσπαθεί να αντιστοιχήσει το ερώτημα σε μια συγκεκριμένη κατηγορία, να προτείνει μια υποκατηγορία ή να εφαρμόσει κάποια facet που υπάρχουν στην κατηγορία και αναφέρονται στα ερωτήματα. Τέτοιες περιπτώσεις συναντώνται πάρα πολύ συχνά σε καταστάσεις όπως αναζήτηση εργασίας, αναζήτηση για επιλογές διασκέδασης, αναζήτηση για σπίτια, αναζήτηση για αγγελίες, αναζήτηση σε ηλεκτρονικά καταστήματα κοκ και μπορεί να έχουν ευρεία εφαρμογή σε πολλές περιπτώσεις. Η διπλωματική θα αναλύσει και θα αναπτύξει τέτοιες τεχνικές ώστε να υλοποιήσει μια τέτοια μηχανή αναζήτησης.

Η διπλωματική εργασία περιλαμβάνει την μελέτη του Elasticsearch, και ιδιαίτερα του faceted search και fielded search. Μελετώνται τεχνικές NLP (Natural Language Processing) όπως NLP, spell check, entity extraction καθώς και η διαδικασία αναζήτησης σε προκαθορισμένες ταξινομίες. Ακόμη θα παρουσιαστούν τεχνικές παραγωγής προτάσεων επέκτασης αναζήτησης σε επιμέρους τμήματα της ταξινομίας.

1.3 Συνεισφορά διπλωματικής

Η συνεισφορά της διπλωματικής εστιάζεται σε τρία σημεία. Το πρώτο σημείο περιλαμβάνει την μελέτη των τεχνικών επεξεργασία φυσικής γλώσσας, και στην συνέχεια την επιλογή του Google Natural Language Processing ως εργαλείου για την υλοποίηση της εφαρμογής. Στο δεύτερο σημείο μελετήσαμε της μηχανές αναζήτησης και ειδικότερα το Elasticsearch, κάνοντας εκτεταμένη χρήση ταξινομιών διαμόρφωση των δεδομένων της εφαρμογής. Στο τρίτο σημείο ολοκληρώσαμε τα δυο επιμέρους συστήματα σε μία ενιαία μηχανή αναζήτησης η οποία κάνει χρήση των τεχνικών των προηγούμενων τομέων ώστε να εξελιγμένες δυνατότητες αναζήτησης.

Αναλυτικότερα στο πρώτο μέρος μελετήσαμε τα παρακάτω.

1. Το θεωρητικό υπόβαθρο της επεξεργασίας φυσικής γλώσσας και ειδικότερα στα επίπεδα της ανάλυσης φυσικής γλώσσας.
2. Στην εξόρυξη πληροφορίας, NER Named Entity Recognition, στην εξαγωγή σχέσεων και εξαγωγή συμβάντων.
3. Τις τεχνικές μηχανικής μάθησης που χρησιμοποιούνται στην επεξεργασία φυσικής γλώσσας.
4. Τα χαρακτηριστικά του Google NLP.
5. Τις δυνατότητες συντακτικής, γλωσσολογικής και συναισθηματικής ανάλυσης του Google NLP. Εμβαθύναμε ακόμη στις δυνατότητες POS Parts of Speech, εξαγωγής οντοτήτων, NER και στα δέντρα εξαρτήσεων.

Στο δεύτερο μέρος μελετήσαμε.

1. Το γνωστικό πεδίο της ανάκτησης πληροφορίας, μελετώντας τις βασικές αρχές.
2. Το γνωστικό πεδίο του faceted search και της αναζήτησης πλήρους κειμένου.
3. Τις ταξινομίες και οντολογίες
4. Την μηχανή αναζήτηση Elasticsearch, καθώς και τις δυνατότητες και τα πλεονεκτήματα της.

Στο τρίτο μέρος μελετήσαμε

1. Την ανάλυση του dataset και την δημιουργία των απαραίτητων ταξινομιών για τη αναπαράσταση του στην μηχανή αναζήτησης.
2. Την δημιουργία του κομματιού της εφαρμογής για την αποστολή των ερωτημάτων στο Google NLP, καθώς και την ερμηνεία των αποκρίσεων.
3. Την Ανάλυση των pattern και την αποστολή του κατάλληλου ερωτήματος στην μηχανή αναζήτησης.

1.4 Οργάνωση κειμένου

Η εργασία αποτελείται από 6 κεφάλαια και οργανώνεται ως εξής:

Στο 1ο κεφάλαιο γίνεται μια εισαγωγή πάνω σε βασικές έννοιες.

Στο κεφάλαιο 2 παρουσιάζονται βασικές έννοιες και εφαρμογές πάνω στην επεξεργασία της φυσικής γλώσσας. Γίνεται μια ιστορική αναδρομή σε αυτή, και στην συνέχεια παρουσιάζονται οι έννοιες της ανάλυσης φυσικής γλώσσας και των επιπέδων της, οι αρχές της εξαγωγής πληροφορίας, των τεχνικών μηχανικής μάθησης που χρησιμοποιούνται στον τομέα. Τέλος παρουσιάζονται τα βασικά πεδία εφαρμογής της επεξεργασίας φυσικής γλώσσας.

Στο κεφάλαιο 3 γίνεται αναφορά στο πεδίο της ανάκτησης πληροφορίας, παρουσιάζονται οι έννοιες της ανάκτησης, σχετικότητας και ακρίβειας καθώς και η αξιολόγηση της ανάκτησης

πληροφορίας. Ακολούθως μελετάμε την ανάκτηση μέσω κατάταξης και το faceted search. Τέλος διερευνούμε το γνωστικό πεδίο των ταξινομιών και οντολογιών.

Στο 4ο κεφάλαιο γίνεται αναφορά και εκτενής μελέτη του Google NLP. Αναλύουμε τις δυνατότητες γλωσσολογικής, συντακτικής, μορφολογικής, συναισθηματικής ανάλυσης που μας παρέχει καθώς επίσης και τις δυνατότητες Part of speech, εξαγωγής οντοτήτων, NER και τα δέντρα συσχετίσεων που παράγει.

Στο 5ο κεφάλαιο μελετάμε το dataset του IMDB, αναφέρονται οι τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής. Μελετάμε την μηχανή αναζήτησης Elasticsearch. Παραθέτουμε την χαρτογράφηση και τις παραμέτρους που χρησιμοποιήσαμε για την δημιουργία του index, τον τρόπο εξαγωγής γεγονότων, οντοτήτων και σχέσεων. Εμβαθύνουμε στον ορθογραφικό έλεγχο, την ασαφή αναζήτηση, επέκταση ερωτημάτων, δημιουργία προτάσεων και στην εξαγωγή προτύπων αναζήτησης. Ακόμη παρουσιάζεται η διεπαφή της εφαρμογής.

Τέλος στο κεφάλαιο 6, παρουσιάζονται τα συμπεράσματα που προκύπτουν μέσα από την ανάπτυξη της εφαρμογής και γίνονται προτάσεις για μελλοντικές βελτιώσεις της.

2

Επεξεργασία Φυσικής Γλώσσας

2.1 Εισαγωγή

Η Επεξεργασία Φυσικής Γλώσσας (Natural Language Processing NLP) αποτελεί κομμάτι της Τεχνητής Νοημοσύνης και της Γλωσσολογίας, αφιερωμένο στην κατανόηση, λεκτικών δηλώσεων ή κειμένων που έχουν γραφεί σε ανθρώπινη γλώσσα από τους υπολογιστές. Η επεξεργασία φυσικής γλώσσας δημιουργήθηκε για να διευκολύνει την εργασία του χρήστη και να ικανοποιήσει την επιθυμία του χρήστη για επικοινωνία με τον υπολογιστή σε φυσική γλώσσα. Δεδομένου πολλοί χρήστες μπορεί να μην είναι εξοικειωμένοι με τις γλώσσες που χρησιμοποιούν οι μηχανές , η Επεξεργασία Φυσικής Γλώσσας, εξυπηρετεί ιδιαιτέρως εκείνους τους χρήστες που δεν έχουν αρκετό χρόνο για να μάθουν νέες γλώσσες μηχανής ή να τελειοποιήσουν την υπάρχουσα γνώση που έχουν σε μία συγκεκριμένη γλώσσα η οποία και χρησιμοποιείται από την εφαρμογή τους.

Μπορούμε να ορίσουμε την γλώσσα μηχανής ως σύνολο κανόνων ή σύνολο συμβόλων. Τα σύμβολα συνδυάζονται και χρησιμοποιούνται για τη μεταφορά ή τη μετάδοση πληροφοριών. Τα σύμβολα συμπληρώνονται από κανόνες. Η επεξεργασία φυσικής γλώσσας μπορεί να ταξινομηθεί σε δύο βασικά μέρη, τη Κατανόηση Φυσική Γλώσσας (Natural Language Understanding) και τη Δημιουργία Φυσικής Γλώσσας (Natural Language Generation), η οποία έχει ως σκοπό την κατανόηση και να δημιουργία κείμενου.

Η γλωσσολογία είναι η επιστήμη της γλώσσας που περιλαμβάνει τη Φωνολογία (Phonology) η οποία αναφέρεται στον ήχο, τη μορφολογία λέξεων (Morphology) η οποία πραγματεύεται τη δομή των λέξεων, τη σύνταξη προτάσεων (Syntax), τη σημασιολογική σύνταξη (Semantics Syntax) και την πραγματολογία (Pragmatics) που αναφέρεται στην κατανόηση και πιο συγκεκριμένα στη διευκρίνιση του περιεχομένου της κεντρικής έννοιας.

Ο Noah Chomsky, ένας από τους πρώτους γλωσσολόγους του εικοστού αιώνα, ο οποίος και ανέδειξε τον τομέα των συντακτικών θεωριών, πρόταξε μια μοναδική θεωρία στον τομέα της

θεωρητικής γλωσσολογίας, η οποία έφερε επανάσταση στον τομέα της συντακτικής ανάλυσης [1]. Η θεωρία του κατηγοριοποιεί σε δύο επίπεδα, ένα υψηλότερου επιπέδου στο οποίο περιλαμβάνεται η αναγνώριση ομιλίας και ένα χαμηλότερο επίπεδο που αντιστοιχεί στη φυσική γλώσσα.

Οι ερευνητικοί τομείς του NLP περιλαμβάνουν την αυτόματη σύνοψη (Automatic Summarization), αποσαφήνιση μέσω συσχέτισης (Co-Reference Resolution), την ανάλυση του λόγου (Discourse Analysis), τη μηχανική μετάφραση (Machine Translation), τη μορφολογική κατάτμηση (Morphological Segmentation), την αναγνώριση κατονομασμένων οντοτήτων (Named Entity Recognition), την αναγνώριση οπτικών χαρακτήρων (Optical Character Recognition), την επισήμανση μέρους του λόγου (Part Of Speech Tagging, POS tagging). Η αυτόματη σύνοψη παράγει μια κατανοητή περίληψη ενός συνόλου κειμένου και παρέχει περιλήψεις ή λεπτομερείς πληροφορίες ενός γνωστού κειμένου. Η αποσαφήνιση μέσω συσχέτισης αναφέρεται σε μια πρόταση ή σε μεγαλύτερο σύνολο κειμένου, που καθορίζει ποιες λέξεις αναφέρονται στο ίδιο αντικείμενο. Η ανάλυση του λόγου αναφέρεται στο έργο της αναγνώρισης της δομής του συσχετισμένου κειμένου.

Η Μηχανική μετάφραση αναφέρεται στην αυτόματη μετάφραση κειμένου από μια ανθρώπινη γλώσσα σε άλλη. Ο Μορφολογικός κατακερματισμός αναφέρεται σε ξεχωριστές λέξεις σε μεμονωμένες μορφές και προσδιορίζει την τάξη των μορφωμάτων. Η αναγνώριση κατονομασμένων οντοτήτων (NER) καθορίζει ποια στοιχεία του κειμένου σχετίζονται με προκαθορισμένα ονόματα. Η αναγνώριση οπτικών χαρακτήρων (OCR) αναλύει κείμενο που αναπαρίσταται με μορφή εικόνας αντιστοιχίζοντας το σε αντίστοιχο ή σχετικό κείμενο.

Η επισήμανση μέρους του λόγου (POS tagging), περιγράφει μια πρόταση προσδιορίζοντας το μέρος λόγου στο οποίο ανήκει η κάθε λέξη. Οι διάφορες εργασίες οι οποίες δομούν την NLP είναι προφανώς πολύ στενά συνδεδεμένες μεταξύ τους και χρησιμοποιούνται συχνά για σε συνδυασμό ώστε να παράξουν καλύτερο αποτέλεσμα. Ορισμένες εργασίες, όπως η αυτόματη σύνοψη, η αποσαφήνιση μέσω συσχέτισης κ.λπ., λειτουργούν ως υποσύνολα που χρησιμοποιούνται στην επίλυση μεγαλύτερων εργασιών.

Ο στόχος της επεξεργασίας φυσικής γλώσσας είναι να διευκολύνει μία ή περισσότερες λειτουργίες ενός αλγορίθμου ή ενός συστήματος.

Το μεγαλύτερο μέρος των εργασιών στην επεξεργασία φυσικής γλώσσας διενεργείται από επιστήμονες υπολογιστών, ενώ διάφοροι επαγγελματίες έχουν δείξει ενδιαφέρον όπως οι γλωσσολόγοι, οι ψυχολόγοι οι φιλόσοφοι κλπ. Μια από τις πιο ειρωνικές πτυχές του NLP είναι ότι προστίθεται στη γνώση της ανθρώπινης γλώσσας. Το επιστημονικό πεδίο της Επεξεργασίας Φυσικής Γλώσσας σχετίζεται με διαφορετικές θεωρίες και τεχνικές που πραγματεύονται το πρόβλημα της επικοινωνίας με χρήση φυσικής γλώσσας με τους υπολογιστές. Η ασάφεια είναι ένα από τα σημαντικότερα προβλήματα της επεξεργασία φυσικής γλώσσας, που συνήθως

αντιμετωπίζεται σε συντακτικό επίπεδο, το οποίο έχει ως κομμάτια του την λεκτική και μορφολογική ανάλυση. Οι τομείς αυτοί ασχολούνται με τη μελέτη των λέξεων και του σχηματισμού των λέξεων.

Κάθε ένα από αυτά τα επίπεδα μπορεί να προκαλέσει ασάφειες που μπορούν να επιλυθούν με τη γνώση του συνόλου της πρότασης. Η ασάφεια μπορεί να λυθεί με διάφορες μεθόδους όπως η ελαχιστοποίηση της ασάφειας, η διαφύλαξη της ασάφειας, η αλληλεπιδραστική αμφισημίας και η στάθμισης ασάφεια [3] [4] [5].

2.2 Ιστορική αναδρομή

Πριν από τη δεκαετία του '70, οι περισσότεροι ερευνητές του NLP επικεντρώνονταν στην μηχανική μετάφραση. Το NLP ήταν μια πολύ πρώιμη εφαρμογή του της επιστήμης των υπολογιστών και άρχισε περίπου την ίδια χρονική στιγμή που ο Chomsky δημοσίευε τα πρώτα του σημαντικά έργα σχετικά με την γλωσσολογία (η γλωσσολογία του Chomskyan έγινε γρήγορα κυρίαρχη, ειδικά στις ΗΠΑ). Τη δεκαετία του 1950 και στις αρχές της δεκαετίας του 1960 αναπτύχθηκαν ιδέες για την επίσημη γραμματική στη γλωσσολογία και αναπτύχθηκαν αλγόριθμοι για την ανάλυση φυσικής γλώσσας ταυτόχρονα με τους αλγορίθμους για την ανάλυση των γλωσσών προγραμματισμού. Ωστόσο, οι περισσότεροι γλωσσολόγοι δεν ενδιαφέρθηκαν για το NLP και η προσέγγιση που ανέπτυξε ο Chomsky αποδείχθηκε ότι ήταν εν μέρη έμμεσα χρήσιμη για την NLP.

Το NLP στη δεκαετία του 1970 και στο πρώτο μισό της δεκαετίας του 1980 βασιζόταν κυρίως σε ένα παράδειγμα όπου η εκτεταμένη γλωσσική και πραγματική γνώση ήταν ουσιαστικά αδύνατον να συνδεθούν. Υπήρξε διαμάχη σχετικά με το πόσο η βαθιά γλωσσική γνώση ήταν απαραίτητη για την επεξεργασία, με ορισμένους ερευνητές να υποβαθμίζουν τη σύνταξη. Οι ερευνητές του NLP στο μεγαλύτερο μέρος τους αποτελούσαν μέλη της κοινότητας της τεχνητής νοημοσύνης (AI) (ειδικά στις Η.Π.Α. και το Η.Β.). Οι εξελίξεις στο πεδίο της τεχνητής νοημοσύνης σχετικά με τη χρήση της λογικής έναντι άλλων παραστατικών εννοιών («σκέτο» έναντι «θρασύ») επηρέασαν επίσης την εξέλιξη της NLP.

Μέχρι τη δεκαετία του 1980, εμφανίστηκαν αρκετοί γλωσσικοί φορμαλισμοί οι οποίοι ήταν πλήρως και επίσημα αιτιολογημένοι και σχετικά δυνατόν να αποδοθούν υπολογιστικά. Δυστυχώς, αυτό δεν οδήγησε σε πολλά χρήσιμα συστήματα, εν μέρει επειδή πολλά από τα δύσκολα προβλήματα (αποσαφήνιση κ.λπ.) θεωρήθηκαν ως εργασία κάποιου άλλου (και η πλειοψηφία της κοινότητας της τεχνητής νοημοσύνης δεν ανέπτυξε επαρκείς τεχνικές αναπαράστασης της γνώσης), εν μέρει επειδή οι περισσότεροι ερευνητές επικεντρώνονταν στις εφαρμογές τύπου «πράκτορα» παραμελώντας την ανάπτυξη εφαρμογών που διευκολύνουν το χρήστη. Παρόλο που τα συμβολικά και τα συστήματα βασισμένα στην γλώσσα, μερικές φορές

δούλεψαν αρκετά καλά για τον προσδιορισμό τη φυσική γλώσσας (Native Language Identification NLID), αποδείχτηκαν ελάχιστα χρήσιμα στην επεξεργασία ελεύθερου κειμένου.

Η στατιστική προσέγγιση του NLP υιοθετήθηκε ευρέως τη δεκαετία του 1990, από την πλειοψηφία της ερευνητικής κοινότητας. Η αναγνώριση ομιλίας απέδειξε ότι οι απλές στατιστικές τεχνικές ήταν λειτουργικές στην πράξη, δεδομένου ότι υπήρχαν αρκετά δεδομένα για την εκπαίδευση των συστημάτων. Δημιουργήθηκαν συστήματα NLP που απαιτούσαν πολύ περιορισμένη γνώση χρήσης κώδικα προγραμματισμού, πέρα από το προγραμματισμό του υλικό της αρχικής εκπαίδευσης του συστήματος.

Οι περισσότερες εφαρμογές ήταν πολύ πιο ανεπτυγμένες από τις προηγούμενες εφαρμογές προσδιορισμού φυσική γλώσσας, αλλά η μετάβαση στη στατιστική προσέγγιση του NLP συνέπεσε με αλλαγές στη χρηματοδότηση των ΗΠΑ, η οποία άρχισε να δίνει έμφαση σε διεπαφές βασισμένες στην ομιλία. Υπήρξε επίσης μια γενική συνειδητοποίηση της σημασίας της σοβαρής αξιολόγησης και της καταγραφής αποτελεσμάτων κατά τρόπο που θα μπορούσε να αναπαραχθεί από άλλους ερευνητές. Η χρηματοδότηση των ΗΠΑ έδωσε έμφαση στους διαγωνισμούς με συγκεκριμένους στόχους και παρείχε το υλικό δοκιμής, το οποίο ενθάρρυνε τους ερευνητές να εστιάσουν σε συγκεκριμένο τομέα αλλά στον αντίποδα είχε το μειονέκτημα ότι ορισμένες από τις τεχνικές που αναπτύχθηκαν ήταν υπέρ εστιασμένες σε συγκεκριμένες εργασίες.

Θα πρέπει να υπογραμμιστεί ότι διεξήχθησαν συνεργατικές δράσεις σε συλλογές δεδομένων για πολλά χρόνια (στο μεγαλύτερο μέρος τους από τους γλωσσολόγους). Οι συνεργατικές δράσεις αυτές διεξήχθησαν στα τέλη της δεκαετίας του 1980, όπου ο χώρος στα αποθηκευτικά μέσα έγινε φθηνός και οι συλλογές ήταν εύκολα προσβάσιμες από τους ερευνητές. Παρά τη μετατόπιση των ερευνητικών εργασιών στις στατιστικές προσεγγίσεις, τα περισσότερα εμπορικά συστήματα βασίστηκαν κατά κύριο λόγο σε δύσχρηστες γλωσσολογικές προσεγγίσεις.

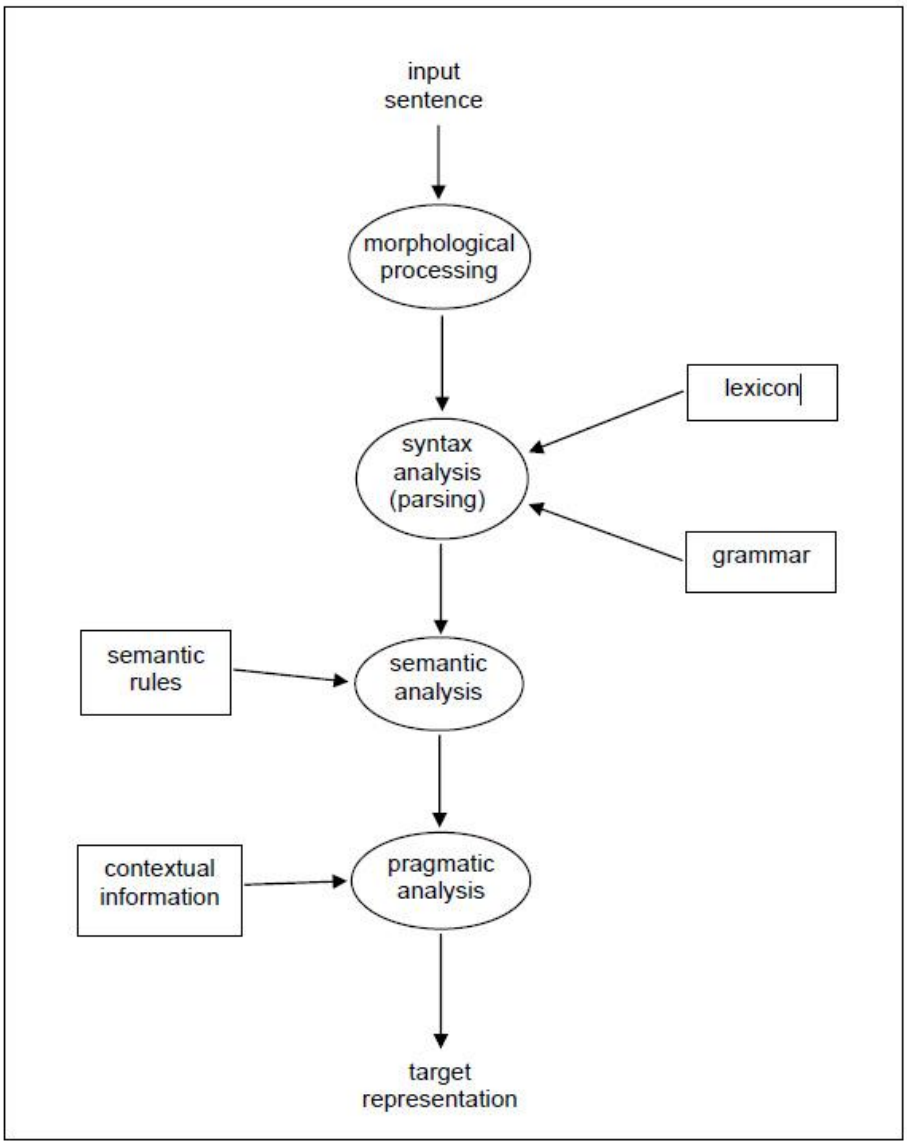
Πιο πρόσφατα, η συμβολική στατιστική διάσπαση έχει γίνει λιγότερο έντονη, καθώς οι περισσότεροι ερευνητές ενδιαφέρονται και για τα δύο. [6] Υπάρχει μεγάλη έμφαση στη μηχανική μάθηση γενικά, συμπεριλαμβανομένης της μηχανικής μάθησης για συμβολική επεξεργασία. Η γλωσσολογικά βασιζόμενη επεξεργασία φυσικής γλώσσας έχει επιστρέψει, με την αυξανόμενη διαθεσιμότητα πώρων ανοιχτού κώδικα και την συνειδητοποίηση ότι τουλάχιστον μερικές από τις κλασσικές στατιστικές τεχνικές φαίνεται να έχουν φτάσει στα όρια των επιδόσεων τους, ιδίως λόγω δυσκολιών προσαρμογής σε νέους τύπους κειμένου. Ωστόσο, οι σύγχρονες γλωσσικές προσεγγίσεις χρησιμοποιούν τη μηχανική μάθηση και τη στατιστική επεξεργασία. Η εκρηκτική άνοδος της χρήσης του διαδικτύου έχει επηρεάσει σημαντικά το NLP, αλλά είναι πολύ νωρίς για να προβλέψουμε το ποιες είναι οι μακροπρόθεσμες συνέπειες. Η πανταχού παρουσία του Διαδικτύου έχει σίγουρα αλλάξει το

χώρο των εφαρμογών NLP και το τεράστιο διαθέσιμο κείμενο μπορεί να αξιοποιηθεί, ιδιαιτέρως από τις στατιστικές τεχνικές.

2.3 Ανάλυση Φυσικής Γλώσσας

2.3.1 Επίπεδα Ανάλυσης

Μια απλοποιημένη άποψη της επεξεργασίας φυσικής γλώσσας τονίζει τέσσερα διαφορετικά στάδια [Εικ. 2.1]. Σε πραγματικά συστήματα αυτά τα στάδια σπάνια συμβαίνουν όλα ως πλήρως διαχωρισμένες, διαδοχικές διαδικασίες. Στην επισκόπηση που ακολουθεί θεωρείται ότι η συντακτική ανάλυση και η σημασιολογική ανάλυση θα αντιμετωπιστούν από τον ίδιο μηχανισμό - τον αναλυτή. Το υπόλοιπο τμήμα αυτής της ενότητας εξετάζει τις διαδικασίες που παρουσιάζονται στο διάγραμμα.



Εικόνα 1 Τα επίπεδα της επεξεργασίας της φυσικής γλώσσας

2.3.1.1 Φωνολογικό Επίπεδο

Η φωνολογία είναι το τμήμα της γλωσσολογίας που αναφέρεται στη συστηματική διάταξη του ήχου. Ο όρος φωνολογία (Phonology) προέρχεται από την αρχαία ελληνική γλώσσα και ο όρος phono - που σημαίνει φωνή ή ήχος, και το επίθεμα - λέξη αναφέρεται σε λέξη ή ομιλία. Το 1993 ο Nikolai Trubetzkoy δήλωσε ότι η φωνολογία είναι «η μελέτη του ήχου που αφορά το σύστημα της γλώσσας» Ο Lass το 1998 έγραψε ότι η φωνολογία αναφέρεται γενικά στους ήχους της γλώσσας, που ασχολούνται με τη σύμπτυξη γλωσσολογικών όρων, ενώ θα μπορούσε να εξηγείται ως εξής: "η φωνολογία είναι η μόνη που ασχολείται με τη λειτουργία, τη συμπεριφορά και την οργάνωση των ήχων ως γλωσσικά στοιχεία. Η φωνολογία περιλαμβάνει τη σημασιολογική χρήση του ήχου για την κωδικοποίηση της σημασίας οποιασδήποτε ανθρώπινης γλώσσας. [6]

2.3.1.2 Μορφολογικό Επίπεδο

Το προκαταρκτικό στάδιο που λαμβάνει χώρα πριν από την ανάλυση σύνταξης είναι η μορφολογική επεξεργασία. Το προκαταρκτικό αυτό στάδιο της επεξεργασίας γλώσσας έχει σκοπό να σπάσει τη διαδικασία εισαγωγής γλώσσας σε ομάδες τμημάτων που αντιστοιχούν σε διακριτές λέξεις, δευτερεύουσες λέξεις και μορφές στίξης. Για παράδειγμα, μια λέξη όπως "unhappily " μπορεί να χωριστεί σε τρεις λέξεις υπό-λέξεων όπως αναλύεται παρακάτω:

Η μορφολογία αφορά κυρίως την αναγνώριση του τρόπου με τον οποίο οι βασικές λέξεις (ρίζες) έχουν τροποποιηθεί για να σχηματίσουν άλλες λέξεις με παρόμοιες έννοιες αλλά συχνά με διαφορετικές συντακτικές κατηγορίες. Η τροποποίηση συμβαίνει συνήθως με την προσθήκη προθεμάτων ή / και επιθεμάτων, επίσης και άλλες αλλαγές κειμένου μπορούν να πραγματοποιηθούν. Γενικά, υπάρχουν τρεις διαφορετικές περιπτώσεις τροποποίησης της μορφής των λέξεων.

Έννοια: η γραφική απεικόνιση των λέξεων αλλάζει λόγω των συντακτικών ρόλων τους. Στην αγγλική γλώσσα, για παράδειγμα, τα περισσότερα ουσιαστικά ονόματα παίρνουν -s ως επίθημα (μπορεί να απαιτούν και άλλες τροποποιήσεις), οι συγκριτικές και υπερθετικές μορφές των απλών επίθετων παίρνουν τα επιθέματα -er και -es.

Απόκλιση: νέες λέξεις προέρχονται από υπάρχουσες λέξεις. Αυτή η κατασκευή λέξεων συμβαίνει συχνά με κανονικό τρόπο επιτρέποντας την τήρηση σαφών μορφολογικών κανόνων. Για παράδειγμα, στα αγγλικά ορισμένα επίθετα παίρνουν -ness ως επίθημα όταν χρησιμοποιούνται για τη δημιουργία ουσιαστικών (happy → happiness). Οι ίδιες αρχές ισχύουν στις περισσότερες ανθρώπινες γλώσσες αν και οι κανόνες είναι διαφορετικοί .

Σύνθεση: δημιουργούνται νέες λέξεις με την ομαδοποίηση των υπαρχουσών λέξεων. Αυτό συμβαίνει σπάνια στα αγγλικά (παραδείγματα περιλαμβάνουν τη λέξη "headache" και

"toothpaste") αλλά χρησιμοποιείται ευρέως και σε άλλες γλώσσες όπου είναι μορφολογικά εφικτό να υπάρχουν άπειρες λέξεις.

Η φύση της μορφολογικής επεξεργασίας εξαρτάται σε μεγάλο βαθμό από τη γλώσσα που αναλύεται. Σε ορισμένες γλώσσες, μεμονωμένες λέξεις (που χρησιμοποιούνται ως ρήματα) περιέχουν όλες τις πληροφορίες σχετικά με τον χρόνο, το πρόσωπο και τον αριθμό μιας φράσης. Σε άλλες γλώσσες, αυτές οι πληροφορίες μπορούν να βρίσκονται διάσπαρτες σε πολλές λέξεις. Για παράδειγμα, η αγγλική πρόταση " I will have been walking..." εμφανίζει σύνθετες χρονικές πληροφορίες οι οποίες είναι διαθέσιμες μετά από την εξέταση της δομής των βοηθητικών ρημάτων. Ορισμένες γλώσσες επισυνάπτουν προθέματα / επίθημα στα ουσιαστικά για να υποδείξουν τους ρόλους τους (βλ. Εικόνα 2.2), άλλες χρησιμοποιούν κλίσεις για να παρέχουν συναφείς πληροφορίες.

Σαν γλώσσα, η αγγλική είναι ευκολότερη από άλλες στην τμηματοποίηση και στην μορφολογική ανάλυση. Σε ορισμένες γλώσσες κυρίως ασιατικές οι λέξεις δεν διαχωρίζονται με απόσταση στη γραπτή τους μορφή (παραδείγματα περιλαμβάνουν ιαπωνικές και ορισμένες κινεζικές γλώσσες). Σε πολλές γλώσσες η μορφολογία των λέξεων μπορεί να είναι διφορούμενη με τρόπους που μπορούν να επιλυθούν μόνο με τη διεξαγωγή συντακτικής και / ή σημασιολογικής ανάλυσης του ζητούμενου κειμένου.

Απλά παραδείγματα στα αγγλικά συμβαίνουν μεταξύ πλήθους ουσιαστικών και ειδικών ρημάτων: "climbs" όπως σε " there are many climbs in the Alps" ή " there are many climbs in the Alps' or 'he climbs Everest in March ". Αυτό το παράδειγμα αμφισημίας μπορεί να επιλυθεί μόνο με συντακτική ανάλυση, θα μπορούσαμε να αναφέρουμε και άλλα παραδείγματα τα οποία θα ήταν ακόμη πιο πολύπλοκα. Το "Undoable" θα μπορούσε να αναλυθεί ως (un-do) -able) ή ως (un- (do-able)), εμφανίζοντας μια ασάφεια η οποία δεν μπορεί πάντα να επιλυθεί μόνο στο επίπεδο σύνταξης.

Το αποτέλεσμα της φάσης της μορφολογικής επεξεργασίας είναι μια σειρά διακριτών λέξεων τα οποία στη συνέχεια μπορούν να χρησιμοποιηθούν για αναζήτηση σε κάποιο λεξικό. Αυτή η σειρά διακριτών λέξεων μπορεί να περιέχει χρόνους, αριθμούς, φύλο και πληροφορίες εγγύτητας, ανάλογα με τη γλώσσα και σε ορισμένες περιπτώσεις μπορεί επίσης να περιέχουν επιπλέον συντακτικές πληροφορίες για τον αναλυτή. Το επόμενο στάδιο επεξεργασίας είναι ανάλυση σύνταξης.

2.3.1.3 Συντακτικό και σημασιολογικό επίπεδο

Ένας επεξεργαστής γλώσσας πρέπει να εκτελεί μια σειρά από διαφορετικές λειτουργίες που βασίζονται κυρίως στην ανάλυση σύνταξης και στη σημασιολογική ανάλυση. Ο σκοπός της ανάλυσης σύνταξης είναι διπλός: να ελέγξουμε ότι μια σειρά λέξεων (μια πρόταση) είναι καλά διαμορφωμένη και να την διασπάσουμε σε μια δομή που δείχνει τις συντακτικές σχέσεις μεταξύ

των διαφόρων λέξεων. Ένας αναλυτής σύνταξης (analyser) πραγματοποιεί την λειτουργία αυτή χρησιμοποιώντας ένα λεξικό ορισμών των λέξεων (lexicon) και ένα σύνολο συντακτικών κανόνων (grammar). Ένα απλό λεξικό περιέχει μόνο τη συντακτική κατηγορία κάθε λέξης, μια απλή γραμματική περιγράφει κανόνες που υποδεικνύουν μόνο πώς μπορούν να συνδυαστούν οι συντακτικές κατηγορίες για να σχηματίσουν φράσεις διαφορετικών τύπων

Παραδείγματα απλού lexicon και grammar θα μπορούσαν να είναι

Lexicon

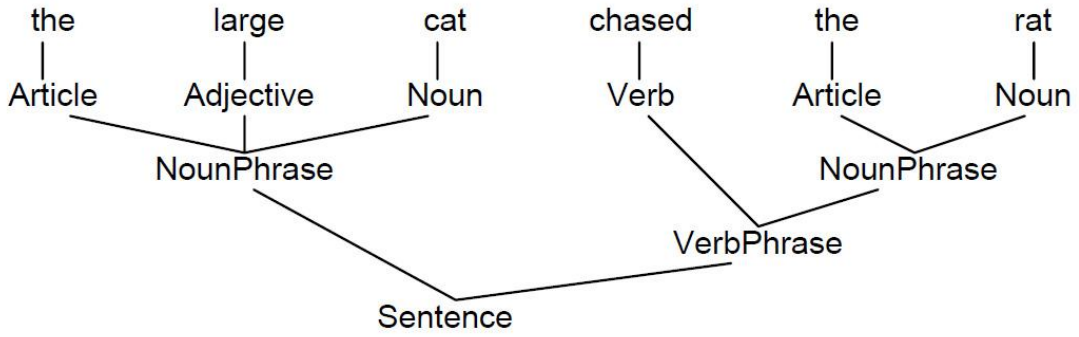
word	category
cat	Noun
chased	Verb
large	Adjective
rat	Noun
the	Article

Grammar

Sentence → NounPhrase, VerbPhrase[2]
VerbPhrase → Verb, NounPhrase
NounPhrase → Article, Noun
NounPhrase → Article, Adjective, Noun

Πίνακας 1 Παράδειγμα απλού λεξικού και γραμματικής

Αυτός ο συνδυασμός γραμματικής-λεξικού θα μπορούσε να αναλύσει την φράση " The large cat chased the rat " ως εξής:



Εικόνα 2Συντακτική ανάλυση της φράσης 'The large cat chased the rat'

Συχνά, η εργασία που επιτελεί ένας επεξεργαστής γλώσσας είναι να αναλύσει μια φράση σε μια γλώσσα όπως η αγγλική και να παράγει μια έκφραση σε κάποια τυποποιημένη σημειογραφία (notation), η οποία, όσον αφορά το σύστημα πληροφορικής, εκφράζει συνοπτικά τη σημασιολογία της φράσης. Μια διεπαφή σε μια βάση δεδομένων μπορεί, για παράδειγμα, να απαιτεί έναν επεξεργαστή γλώσσας για να μετατρέπει τις προτάσεις στα αγγλικά ή τα γερμανικά σε ερωτήματα SQL. Η σημασιολογική ανάλυση είναι ο όρος που δίνεται στην παραγωγή αυτής της τυποποιημένης σημασιολογικής αντιπροσώπευσης.

Προκειμένου να διεξαχθεί σημασιολογική ανάλυση, το λεξικό πρέπει να επεκταθεί ώστε να περιλαμβάνει σημασιολογικούς ορισμούς για κάθε λέξη που περιέχει και η γραμματική πρέπει να επεκταθεί για να διευκρινίσει πώς σχηματίζεται η σημασιολογία οποιασδήποτε φράσης από τη σημασιολογία των επιμέρους στοιχείων της. Για παράδειγμα, ο παραπάνω κανόνας γραμματικής VerbPhrase → Verb, NounPhrase δηλώνει πως η συντακτική ομάδα που ονομάζεται VerbPhrase σχηματίζεται από άλλες συντακτικές ομάδες αλλά δεν περιέχει στοιχεία για τη σημασιολογία οποιουδήποτε προκύπτοντος VerbPhrase. Χρησιμοποιώντας μια απλοποιημένη μορφή λογικής η γραμματική και το λεξικό μπορεί να επεκταθεί για να συλλάβει μερικές σημασιολογικές πληροφορίες. Αυτό απεικονίζεται στο ακόλουθο παράδειγμα.

Lexicon

word	category	semantics
cat	Noun	λx • feline(x)
chased	Verb	λxy • x Λ y Λ chased(x, y)
large	Adjective	λx • largesize(x)
rat	Noun	λx • rodent(x)
the	Article	$\exists_1$ ⟨gensym⟩

Πίνακας 2 Παράδειγμα πρωτογενούς σημασιολογικής ανάλυσης

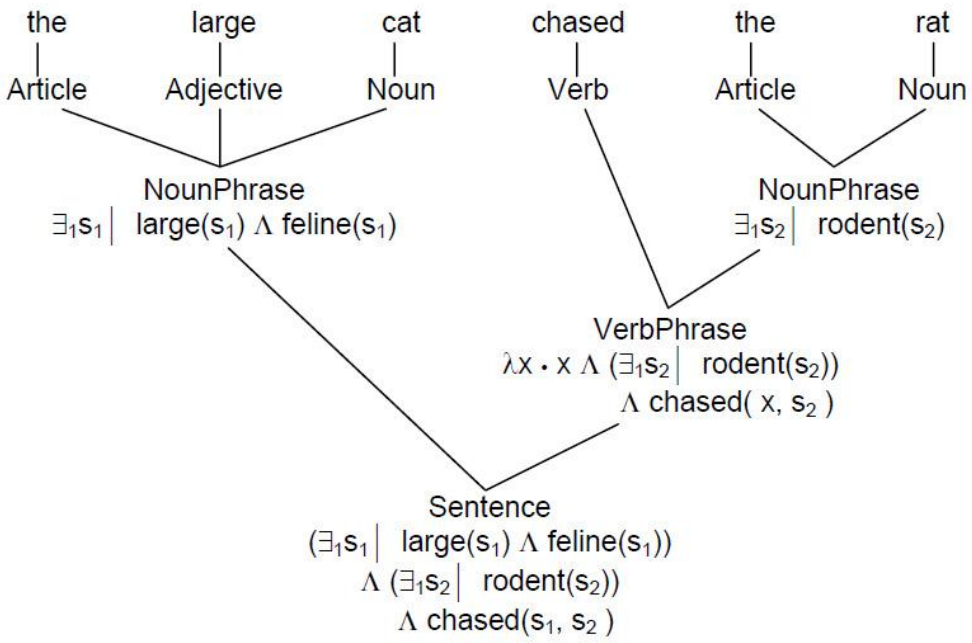
Σημείωση: Οι λέξεις "feline" και "chased" χρησιμοποιούνται σε αυτό το παράδειγμα σαν πρωταρχικές σημασιολογικές σχέσεις.

Grammar

Syntactic rule	Semantic rule
Sentence → NounPhrase, VerbPhrase	apply VerbPhrase (NP)
VerbPhrase → Verb, NounPhrase	apply Verb (NounPhrase)
NounPhrase → Article, Noun	apply Noun (Article)
NounPhrase → Article, Adjective, NounPhrase	apply Adjective (Article) Λ apply Noun (Article)

Πίνακας 3 Παράδειγμα 2 πρωτογενούς σημασιολογικής ανάλυσης

Σημείωση: τα παραπάνω είναι απλοποιημένα για την αναγνωσιμότητα, το apply χρησιμοποιείται για να παράσχει ένα επιχείρημα σε μια μορφή: apply lx • rodent (x) (Ralf) → rodent (Ralf)



Εικόνα 3 Συντακτικές και σημασιολογικές δομές που παράγονται από την ανάλυση της φράσης 'The large cat chased the rat'

Το παραπάνω παράδειγμα δείχνει πώς χρησιμοποιούνται οι κανόνες γραμματικής για να καθορισθεί η διαδικασία παραγωγής σημασιολογικών μορφών. Ο κανόνας που περιγράφει τη σωστή συντακτική μορφή για ένα VerbPhrase, για παράδειγμα, περιγράφει επίσης τον τρόπο δημιουργίας σημασιολογίας για φράσεις ρήματος. Με αυτό τον τρόπο σχηματίζονται και ομαδοποιούνται σημασιολογίες σε μεγαλύτερες υπό-φράσεις μέχρις ότου, μετά την εφαρμογή όλων των σχετικών κανόνων, παραχθεί μια σημασιολογική έκφραση που περιγράφει ολόκληρη την πρόταση.

2.3.1.4 Σημασιολογικό και πραγματολογικό Επίπεδο

Μετά τη σημασιολογική ανάλυση το επόμενο στάδιο της επεξεργασίας ασχολείται με την πραγματολογία (Pragmatics). Δυστυχώς δεν υπάρχει γενικά αποδεκτή διάκριση μεταξύ σημασιολογίας και πραγματολογία. Οι περισσότεροι συγγραφείς [8] κάνουν τη διάκριση ως εξής: η σημασιολογική ανάλυση συσχετίζει την έννοια με μεμονωμένες δηλώσεις / προτάσεις. η πραγματολογική ανάλυση ερμηνεύει τα αποτελέσματα της σημασιολογικής ανάλυσης από την οπτική ενός συγκεκριμένου πλαισίου (το πλαίσιο του διαλόγου ή της κατάστασης του κόσμου κ.λπ.). Αυτό σημαίνει ότι με μια φράση όπως η "The large cat chased the rat" η σημασιολογική ανάλυση μπορεί να παράγει μια έκφραση που σημαίνει μεγάλη γάτα αλλά δεν μπορεί να πραγματοποιήσει το περαιτέρω βήμα του συμπεράσματος που απαιτείται για να αναγνωρίσει τη μεγάλη γάτα όπως Felix. Αυτό θα αφεθεί στην πραγματολογική ανάλυση. Σε

ορισμένες περιπτώσεις, όπως το παράδειγμα που μόλις περιεγράφηκε, η πραγματολογική ανάλυση απλώς αντιστοιχεί τα πραγματικά αντικείμενα / συμβάντα που υπάρχουν σε ένα δεδομένο πλαίσιο με αναφορές αντικειμένων που λαμβάνονται κατά τη σημασιολογική ανάλυση. Σε άλλες περιπτώσεις, η πραγματολογική ανάλυση μπορεί να αποσαφηνίσει φράσεις που δεν μπορούν να αποσαφηνιστούν πλήρως κατά τη διάρκεια των φάσεων σύνταξης και σημασιολογικής ανάλυσης. Για παράδειγμα, εξετάστε τη φράση " Put the apple in the basket on the shelf ". Υπάρχουν δύο σημασιολογικές ερμηνείες για αυτή την πρόταση. Χρησιμοποιώντας μια μορφή λογικής για τη σημασιολογία:

1. put the apple which is currently in the basket onto the shelf

$$(\exists_1 a : \text{apple} \mid \exists b : \text{basket} \wedge \text{inside}(a, b)) \wedge \exists_1 s : \text{shelf} \Rightarrow \text{puton}(a, s)$$

2. put the apple into the basket which is currently on the shelf

$$\exists_1 a : \text{apple} \wedge (\exists_1 b : \text{basket} \mid \exists_1 s : \text{shelf} \wedge \text{on}(b, s)) \Rightarrow \text{putin}(a, b)$$

Η πραγματολογική ανάλυση, σύμφωνα με το τρέχον πλαίσιο, θα μπορούσε να επιλέξει μεταξύ των δύο παραπάνω δυνατοτήτων με βάση τις καταστάσεις και τις θέσεις των αντικειμένων στον μέρος του λόγου.

2.3.1.5 Ομιλία (Discourse)

Ενώ η σύνταξη και η σημασιολογία λειτουργούν με επίπεδο πρότασης, το επίπεδο λόγου του NLP λειτουργεί με μονάδες κειμένου μεγαλύτερες από μια πρόταση. Δηλαδή, δεν ερμηνεύει τη σημασία πολλαπλών κειμένων ως απλά συνενωμένες προτάσεις, καθεμία από τις οποίες μπορεί να ερμηνευτεί μεμονωμένα. Αντίθετα, ο λόγος επικεντρώνεται στις ιδιότητες του κειμένου ως συνόλου, που μεταφέρουν νόημα δημιουργώντας συνδέσεις μεταξύ των συνιστωσών προτάσεων. Σε αυτό το επίπεδο μπορούν να εμφανιστούν διάφοροι τύποι επεξεργασίας του λόγου. Δύο από τους πιο συνηθισμένους είναι η ανάλυση της αναφώνησης και η αναγνώριση της δομής του λόγου / κειμένου. Η ανάλυση αναφορών (Anaphora) είναι η αντικατάσταση λέξεων όπως οι αντωνυμίες, οι οποίες από μόνες τους είναι κενές σημασιολογικά, με την κατάλληλη οντότητα στην οποία αναφέρονται [9]. Η αναγνώριση ομιλίας / διάρθρωσης κειμένου καθορίζει τις λειτουργίες των προτάσεων στο κείμενο, οι οποίες, με τη σειρά τους, προσθέτουν επιπρόσθετη πληροφορία στην ουσιαστική αναπαράσταση του κειμένου. Για παράδειγμα, τα άρθρα εφημερίδων μπορούν να αποδιαμορφωθούν σε συστατικά του λόγου όπως: : Lead, Main, Story, Previous Events, Evaluation, Attributed Quotes, and Expectation.

2.3.1.6 Πραγματολογικό Επίπεδο

Η πραγματιστική ασχολείται με τη σταθερή χρήση της γλώσσας σε καταστάσεις, και χρησιμοποιεί το νόημα των κειμένων για να κατανοήσει το στόχο και να εξηγήσει πώς το επιπλέον νόημα διαβάζεται στα κείμενα . Αυτό απαιτεί μεγάλη γνώση του κόσμου, συμπεριλαμβανομένης της κατανόησης των προθέσεων, των σχεδίων και των στόχων. [7].

2.4 Εξόρυξη πληροφορίας IE

2.4.1 Ορισμός

Ο σκοπός της Εξόρυξης Πληροφοριών (Information Extraction IE) είναι να προσδιορίσει τις περιπτώσεις αναφοράς μιας συγκεκριμένης κατηγορίας οντοτήτων, σχέσεων και γεγονότων σε κείμενα φυσικής γλώσσας και την εξαγωγή των σχετικών ιδιοτήτων και τις παραμέτρους αυτών, των προσδιορισμένων οντοτήτων, σχέσεων ή γεγονότων. Οι πληροφορίες που εξάγονται είναι προκαθορισμένες σε δομές που ορίζονται από το χρήστη, οι οποίες ονομάζονται πρότυπα (ή αντικείμενα), το καθένα από τα οποία αποτελείται από ένα αριθμό υποδοχών (ή χαρακτηριστικών). Τα δεδομένα αυτά δημιουργούνται συστήματα εξόρυξης πληροφοριών καθώς εκείνα επεξεργάζονται το κείμενο.

Οι τιμές των χαρακτηριστικών είναι συνήθως, στοιχεία από το κείμενο, με προκαθορισμένες τιμές ή μια αναφορά σε ένα πρότυπο που δημιουργήθηκε προηγουμένως. Ένας τρόπος ερμηνείας ενός συστήματος IE είναι ως ένα σύνολο στοιχείων μιας βάσης δεδομένων, αφού ένα σύστημα IE δημιουργεί μια δομημένη αναπαράσταση επιλεγμένων πληροφοριών που αντλούνται από το κείμενο που αναλύεται. [10][11]

2.4.2 Περιπτώσεις εξόρυξης πληροφορίας

Η εφαρμογή της εξαγωγής πληροφοριών στο κείμενο στοχεύει στη δημιουργία δομημένης άποψης δηλαδή την αναπαράσταση των πληροφοριών που είναι κατανοητές από τη μηχανή. Οι κλασικές λειτουργίες των συστημάτων IE περιλαμβάνουν:

2.4.2.1 Αναγνώριση ονομασίας οντοτήτων

Η Αναγνώριση ονοματοδοτιμένων οντοτήτων Named Entity Recognition (NER) αντιμετωπίζει το πρόβλημα της αναγνώρισης (ανίχνευσης) και της ταξινόμησης προκαθορισμένων τύπων ονομάτων, όπως είναι οι οργανισμοί (π.χ. «Παγκόσμια Οργάνωση Υγείας»), τα πρόσωπα (π.χ. Muammar Kaddafi) τοποθεσιών (π.χ. «Βαλτική Θάλασσα»),

χρονικές εκφράσεις (π.χ. «1η Σεπτεμβρίου 2011»), αριθμητικές και νομισματικές εκφράσεις (π.χ. «20 εκατομμύρια ευρώ») κλπ.

Η λειτουργία της NER μπορεί επιπλέον να περιλαμβάνει την εξαγωγή περιγραφικών πληροφοριών από το κείμενο σχετικά με τις εξαγόμενες οντότητες, μέσω της πλήρωσης ενός πρότυπου μικρής κλίμακας. Για παράδειγμα, στην περίπτωση προσώπων, μπορεί να περιλαμβάνει την εξαγωγή του τίτλου, της θέσης, της εθνικότητας, του φύλου και άλλων χαρακτηριστικών του προσώπου. Είναι σημαντικό να σημειωθεί ότι η NER συνεπάγεται επίσης τη ληματοποίηση (normalisation) των οντοτήτων που εξάγει, κάτι το οποίο είναι ιδιαίτερα σημαντικό σε εξαιρετικά ευμετάβλητες γλώσσες.

2.4.2.2 Ανάλυση μέσω συσχέτισης

Η ανάλυση μέσω συσχέτισης (Co-reference Resolution CO) απαιτεί τον προσδιορισμό πολλαπλών αναφορών (coreferring) της ίδιας οντότητας στο κείμενο. Η αναφορά των οντοτήτων μπορεί να είναι:

- **Pronominal:** (Αντωνυμική) Σε περίπτωση ονομασίας μιας οντότητας. π.χ. «General Electric» και «GE» μπορούν να αναφέρονται στην ίδια οντότητα του πραγματικού κόσμου,
- **Προκαθορισμένη:** Σε περίπτωση που μια οντότητα αναφέρεται με αντωνυμία. π.χ., Ο John αγόρασε τρόφιμα. Αλλά ξέχασε να αγοράσει ποτά », η αντωνυμία αναφέρεται στον John,
- **Ονομαστική:** Σε περίπτωση που μια οντότητα αναφέρεται σε ονομαστική φράση. π.χ., Η Microsoft αποκάλυψε τα κέρδη της. Η εταιρία παρουσίασε επίσης μελλοντικά σχέδια. Η οριστική φράση ουσιαστικών 'Η' εταιρεία αναφέρεται στη Microsoft, και
- **Τυπική:** όπως στην περίπτωση μηδενικής αναφοράς

2.4.2.3 Εξαγωγή σχέσεων

Η εξαγωγή σχέσεων Relation Extraction (RE) ανιχνεύει και ταξινομεί, προκαθορισμένες σχέσεις μεταξύ οντοτήτων που προσδιορίζονται στο κείμενο. Για παράδειγμα:

- Εργαζόμενος (Steve Jobs, Apple): μια σχέση μεταξύ ενός ατόμου και ενός οργανισμού, που εξάγεται από το "Jobs Steve Jobs για την Apple"
- Location In (Smith, New York): μια σχέση μεταξύ ενός ατόμου και ενός τόπου, που προέρχεται από τον κ. Smith έκανε μια ομιλία στο συνέδριο στη Νέα Υόρκη »,
- Θυγατρική (TVN, ITI Holding): μια σχέση μεταξύ δύο εταιρειών που εξήχθη από την εταιρεία TVN που δηλώνει ότι η μητρική της εταιρεία, ITI Holdings, εξετάζει διάφορες επιλογές για την ενδεχόμενη πώληση.

Παρόλο που το σύνολο των σχέσεων, οι οποίες μπορεί να ενδιαφέρουν είναι απεριόριστες, το σύνολο των σχέσεων, σε μια συγκεκριμένη περίπτωση είναι προκαθορισμένο και σταθερό, ως μέρος της εξειδίκευσης του έργου.

2.4.2.4 Εξόρυξη Συμβάντων

Η Εξόρυξη Συμβάντων (Event Extraction EE) αναφέρεται στις διαδικασίες εντοπισμού γεγονότων στο ελεύθερο κείμενο και στην παραγωγή λεπτομερών και δομημένων πληροφοριών σχετικά με αυτές. Ιδανικά προσδιορίζοντας ποιος έκανε τι σε ποιον, πότε, πού, με ποιες μεθόδους (μέσα) και γιατί. Συνήθως, η εξαγωγή συμβάντων περιλαμβάνει την εξαγωγή πολλών οντοτήτων και σχέσεων μεταξύ τους. Για παράδειγμα, η εξαγωγή πληροφοριών σχετικά με τις τρομοκρατικές επιθέσεις από το τμήμα κειμένου 'Masked gunmen armed with assault rifles and grenades attacked a wedding party in mainly Kurdish southeast Turkey, killing at least 44 people.' περιλαμβάνει την ταυτοποίηση των δραστών (Masked gunmen), των θυμάτων αριθμός ανθρώπων που σκοτώθηκαν / τραυματίστηκαν (least 44), όπλα και μέσα που χρησιμοποιήθηκαν (assault rifles and grenades) και τοποθεσία (southeast Turkey). Ένα άλλο παράδειγμα είναι η εξαγωγή πληροφοριών σχετικά με νέες κοινοπραξίες, όπου ο στόχος είναι να προσδιοριστούν οι εταίροι, τα προϊόντα, τα κέρδη και η κεφαλαιοποίηση της κοινοπραξίας. Το EE θεωρείται ότι είναι η πιο δύσκολη από τις τέσσερις περιπτώσεις εξόρυξης πληροφορίας. Ενώ κατά τα πρώτα χρόνια η εξόρυξης πληροφορίας επικεντρώθηκε στην επίλυση προβλημάτων που αναφέρθηκαν παραπάνω σε επίπεδο εγγράφου, η έρευνα έχει μετατοπιστεί στην εξαγωγή πληροφοριών από διασταυρούμενες πηγές πληροφοριών.

2.4.3 Αρχιτεκτονική, στοιχεία συστημάτων εξόρυξης πληροφορίας

Παρόλο που τα συστήματα εξόρυξης πληροφοριών έχουν κατασκευαστεί για διαφορετικές λειτουργίες και διαφέρουν σημαντικά μεταξύ τους, ωστόσο υπάρχουν ορισμένα βασικά στοιχεία τα οποία είναι κοινά μεταξύ τους. Η συνολική αλυσίδα επεξεργασίας IE μπορεί να αναλυθεί σε διάφορες διαστάσεις. Η αλυσίδα περιλαμβάνει συνήθως βασικά γλωσσικά συστατικά τα οποία μπορούν να προσαρμοστούν ή να είναι χρήσιμα για τις λειτουργίες της NLP γενικότερα, καθώς και για στοιχεία που σχετίζονται με την IE τα οποία εκτελούν βασικές λειτουργίες της εξόρυξης πληροφοριών. Η προαναφερθείσα αλυσίδα τυπικά περιλαμβάνει στοιχεία ανεξάρτητα από τον συγκεκριμένο γνωστικό πεδίο, καθώς και στοιχεία που σχετίζονται με το γνωστικό στοιχείο. Το τμήμα που είναι ανεξάρτητο του γνωστικού πεδίου συνήθως αποτελείται από συγκεκριμένα γλωσσικά συστατικά, προκειμένου να εξαχθεί όσο το δυνατόν περισσότερη γλωσσική δομή.[12] Για την εξαγωγή της γλωσσικής δομής συνήθως, εκτελούνται τα ακόλουθα βήματα:

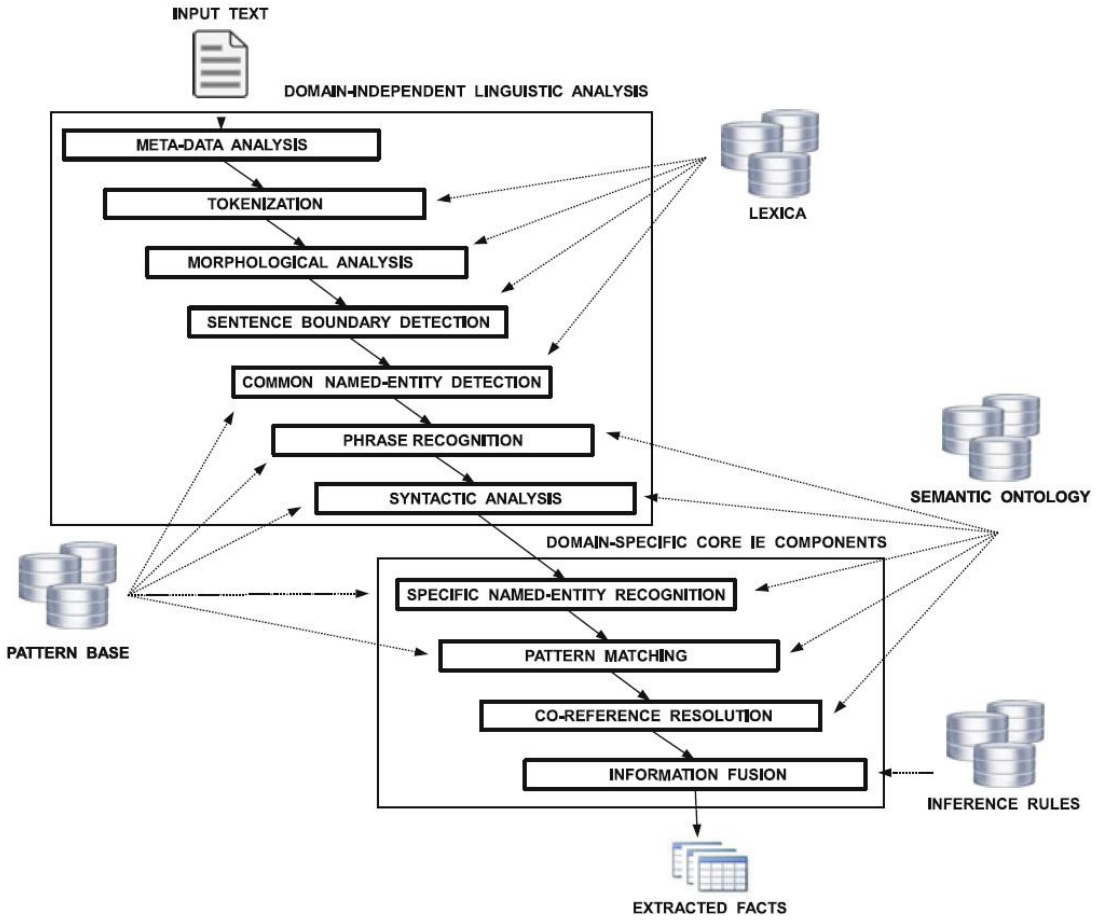
- **Ανάλυση μετα-δεδομένων(Meta-data):** Εξαγωγή του τίτλου, του σώματος, της δομής του σώματος (προσδιορισμός των παραγράφων) και της ημερομηνίας του εγγράφου.
- **Τμηματοποίηση (Tokenization):** Κατάτμηση του κειμένου σε μονάδες επιπέδου λέξης, ονομαζόμενες tokens και ταξινόμηση του τύπου τους, π.χ., αναγνώριση κεφαλαιοποιημένων λέξεων, λέξεις γραμμένες με πεζά γράμματα, λέξεις με συλλαβή, σημεία στίξης, αριθμοί κ.λπ.
- **Μορφολογική ανάλυση:** Εξόρυξη μορφολογικών πληροφοριών από tokens που αποτελούν πιθανές μορφές λέξεων ή τη ρίζες λέξεων (lemma), part of speech, άλλες μορφολογικές ετικέτες ανάλογα με το μέρος του λόγου: π.χ., τα ρήματα έχουν χαρακτηριστικά όπως η χρόνο, πρόσωπο, κλπ. Οι λέξεις που είναι διφορούμενες, μπορούν να αποσαφηνιστούν με βάση τη σχέση τους με ορισμένες μορφολογικές κατηγορίες. Συνήθως γίνεται αποσαφήνιση με βάση με part-of-speech. Ανίχνευση ορίων / καταλήξεων: Τμηματοποίηση κειμένου σε ακολουθία προτάσεων ή δηλώσεων, εκάστη των οποίων αναπαρίσταται ως ακολουθία λεξικών στοιχείων μαζί με τα χαρακτηριστικά τους.
- **Κοινή εξαγωγή ονομαστικής οντότητας (Common Named-entity extraction):** Ανίχνευση οντοτήτων που δεν σχετίζονται με το γνωστικό πεδίο, όπως χρονικές εκφράσεις, αριθμοί και νόμισμα, γεωγραφικές αναφορές κλπ.
- **Αναγνώριση φράσεων:** Αναγνώριση τοπικών δομών μικρής κλίμακας, όπως ουσιαστικά που αποδίδονται σε φράσεις, ομάδες ρημάτων, φράσεις προθέσεων, ακρωνύμια και συντμήσεις.
- **Συντακτική ανάλυση:** Υπολογισμός μιας δομής εξάρτησης (parse tree) της πρότασης με βάση την ακολουθία των λεξικών στοιχείων και των δομών μικρής κλίμακας. Η συντακτική ανάλυση μπορεί να είναι βαθιά ή ρηχή. Στην πρώτη περίπτωση, κάποιος ενδιαφέρεται να υπολογίσει όλες τις πιθανές ερμηνείες (parse trees) και γραμματικές σχέσεις μέσα στην πρόταση. Στην τελευταία περίπτωση, η ανάλυση περιορίζεται στην ταυτοποίηση μη αναδρομικών δομών ή δομών με περιορισμένη ποσότητα δομικής αναδρομής, οι οποίες μπορούν να αναγνωριστούν με υψηλό βαθμό βεβαιότητας και τα γλωσσικά φαινόμενα που προκαλούν προβλήματα (ασάφειες) δεν αντιμετωπίζονται και εκπροσωπούνται με μη δομημένες δομές.

Η έκταση της επεξεργασίας των διαφόρων κειμένων, που δεν συσχετίζονται με κάποιο γνωστικό πεδίο, μπορεί να ποικίλει ανάλογα με τις απαιτήσεις της συγκεκριμένης εφαρμογής. Οι βασικές λειτουργίες της IE-NER, δηλαδή η ανάλυση της συν-παραπομπής και ο εντοπισμός σχέσεων και συμβάντων συνήθως εξειδικεύονται στον συγκεκριμένο τομέα αλλά και υποστηρίζονται από στοιχεία και πόρους του συγκεκριμένου τομέα. Η επεξεργασία κειμένων

που έχουν κοινό γνωστικό πεδίο υποστηρίζεται συνήθως σε χαμηλότερο επίπεδο ανιχνεύοντας εξειδικευμένους όρους σε κείμενο. Για παράδειγμα, σε πεδία που σχετίζονται με την ιατρική, θα είναι απαραίτητα εκτεταμένα εξειδικευμένα λεξικά, οντολογίες και θησαυροί ιατρικών όρων, ενώ για την IE σε τομείς που σχετίζονται με τις επιχειρήσεις, είναι περιττοί.

Μια τυπική αρχιτεκτονική ενός συστήματος IE απεικονίζεται στην εικόνα 4. Στη διαδικασία επεξεργασία κειμένων που έχουν κοινό γνωστικό πεδίο, εφαρμόζεται ένα στοιχείο NER για τον προσδιορισμό των οντοτήτων που σχετίζονται με το συγκεκριμένο γνωστικό πεδίο. Στη συνέχεια, μπορούν να εφαρμοστούν μοτίβα για: (α) προσδιορισμό τμημάτων κειμένου, τα οποία περιγράφουν τις σχέσεις στόχων και τα γεγονότα και (β) εξαγάγουν τα χαρακτηριστικά κλειδιά για να γεμίσουν τις θυρίδες (Slots) του πρότυπου που αναπαριστά τη σχέση / συμβάν.

Ένα στοιχείο συν-αναφοράς προσδιορίζει αναφορές που αναφέρονται στην ίδια οντότητα. Τέλος, τα μερικώς γεμάτα πρότυπα συγχωνεύονται και επικυρώνονται χρησιμοποιώντας εξειδικευμένους κανόνες συμπερασμάτων για να δημιουργηθούν πλήρεις περιγραφές σχέσεων / γεγονότων. Το τελευταίο βήμα είναι κρίσιμο, καθώς οι σχετικές πληροφορίες ενδέχεται να διαχέονται σε διαφορετικές προτάσεις ή ακόμα και σε έγγραφα. Είναι σημαντικό να σημειωθεί ότι στην πράξη τα όρια μεταξύ των στοιχείων γλωσσολογικής ανάλυσης ανεξάρτητων γνωστικών πεδίων και των συστατικών IE πυρήνα μπορεί να είναι θολά, π.χ., μπορεί να υπάρχει ένα μοναδικό συστατικό NER το οποίο εκτελεί ταυτόχρονα NER κοινού και ανεξάρτητου γνωστικού πεδίου. Υπάρχουν πολλά πακέτα λογισμικού, διαθέσιμα τόσο για ερευνητικούς σκοπούς όσο και για εμπορική χρήση, τα οποία παρέχουν διάφορα εργαλεία που μπορούν να χρησιμοποιηθούν στη διαδικασία ανάπτυξης ενός συστήματος IE, που κυμαίνονται από βασικές γλωσσικές μονάδες επεξεργασίας (π.χ., ανιχνευτές γλώσσας, διαιρέτες παραγράφων) μέχρι σε γενικά πλαίσια NLP προσανατολισμένα σε IE.



Εικόνα 4 Τυπική αρχιτεκτονική ενός συστήματος IE

2.5 Μηχανική μάθηση στο NLP

Η στατιστική και η μηχανική μάθηση περιλαμβάνουν την ανάπτυξη (ή τη χρήση) αλγορίθμων που επιτρέπουν σε ένα πρόγραμμα να συνάγει μοτίβα για δεδομένα παραδείγματα «εκπαίδευσης», τα οποία με τη σειρά τους, επιτρέπουν να «γενικεύουν» και να κάνουν προβλέψεις για νέα δεδομένα. Κατά τη διάρκεια της φάσης εκπαίδευσης, οι αριθμητικές παράμετροι που χαρακτηρίζουν ένα υποκείμενο μοντέλο ενός συγκεκριμένου αλγορίθμου υπολογίζονται βελτιστοποιώντας ένα αριθμητικό μέτρο, συνήθως μέσω μιας επαναληπτικής διαδικασίας. [13]

Η εκπαίδευση μπορεί να εποπτεύεται - κάθε στοιχείο των δεδομένων εκπαίδευσης σημειώνεται με τη σωστή απάντηση - ή να μην εποπτεύεται, όπου δεν συμβαίνει το παραπάνω. Η διαδικασία εκπαίδευσης προσπαθεί να αναγνωρίσει αυτόματα τα πρότυπα (όπως στην ανάλυση συμπλεγμάτων και παραγόντων). Μια παγίδα σε οποιαδήποτε προσέγγιση μάθησης είναι η πιθανότητα υπερβολικής προσαρμογής όπου το μοντέλο μπορεί να ταιριάζει σχεδόν πλήρως

στα παραδείγματα δεδομένων, αλλά να κάνει κακές προβλέψεις για νέες περιπτώσεις που δεν έχει ποτέ συναντήσει. Αυτό οφείλεται στο γεγονός ότι μπορεί να μάθει τον τυχαίο θόρυβο στα δεδομένα εκπαίδευσης και όχι μόνο τα ουσιώδη, επιθυμητά χαρακτηριστικά του. Ο κίνδυνος over-fitting ελαχιστοποιείται με τεχνικές όπως το cross-validation, η οποία χωρίζει τυχαία τα παραδείγματα δεδομένων σε εκπαιδευτικά και δοκιμαστικά σετ για την εσωτερική επικύρωση των προβλέψεων του μοντέλου. Αυτή η διαδικασία καταμερισμού, εκπαίδευσης και επικύρωσης δεδομένων επαναλαμβάνεται σε διάφορους γύρους και τα αποτελέσματα επικύρωσης υπολογίζονται κατά μέσο όρο σε όλους τους γύρους.

Τα μοντέλα μηχανικής μάθησης μπορούν γενικά να ταξινομηθούν είτε ως γενικευμένα είτε ως διακριτικά (discriminative). Οι μέθοδοι γενίκευσης επιδιώκουν να δημιουργήσουν πλούσια μοντέλα κατανομών πιθανοτήτων και ονομάζονται έτσι επειδή, με τέτοια μοντέλα, μπορούν να δημιουργηθούν συνθετικά δεδομένα. Οι διακριτικές μέθοδοι είναι πιο δημοφιλής λόγο της δυνατότητας να εκτιμήσουν άμεσα τις μεταγενέστερες πιθανότητες βάσει των παρατηρήσεων.

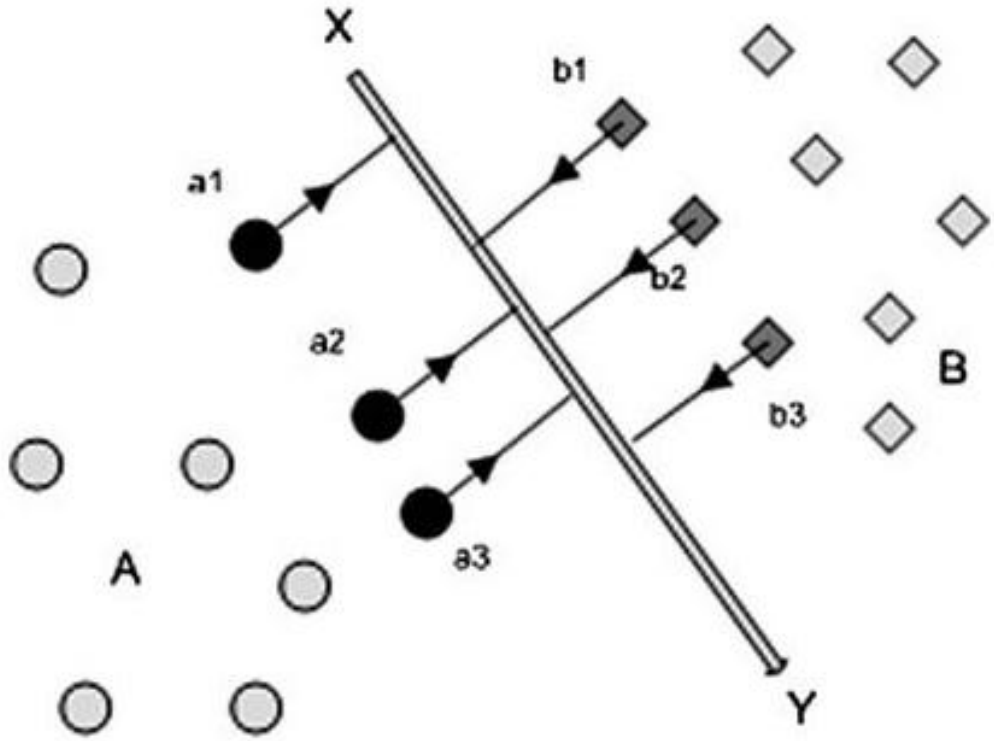
Παραθέτουμε ένα παράδειγμα προς κατανόηση των παραπάνω. Για να εντοπιστεί μια γλώσσα ενός άγνωστου ομιλητή, οι γενικευτικές προσεγγίσεις θα εφαρμόσουν βαθιά γνώση πολλών γλωσσών για να εκτελέσουν την αντιστοίχιση. Οι διακριτικές μέθοδοι βασίζονταν σε μια λιγότερο εξαρτημένη σε γνώση προσέγγιση της χρήσης διαφορών γλωσσών για να βρεθεί η πλησιέστερη αντιστοιχία. Σε σύγκριση με τα μοντέλα γενίκευσης που μπορούν να καταστούν ανυπόφορα όταν χρησιμοποιούνται πολλά χαρακτηριστικά, τα διακριτικά μοντέλα συνήθως επιτρέπουν τη χρήση περισσότερων χαρακτηριστικών. Η Logistic regression και τα conditional random fields (CRFs) είναι παραδείγματα διακριτικών μεθόδων, ενώ οι ταξινομητές Naive Bayes και τα κρυφά μοντέλα Markov (hidden Markov models HMMs) είναι παραδείγματα μεθόδων γενίκευσης.

Ορισμένες κοινές μέθοδοι μηχανικής μάθησης που χρησιμοποιούνται σε εργασίες NLP και χρησιμοποιούνται από πολλά άρθρα σε αυτό το τεύχος συνοψίζονται παρακάτω.

2.5.1.1 Support vector machines (SVMs)

Οι SVM, αποτελούν μια προσέγγιση διακριτικής μάθησης, ταξινομώντας τις εισροές (π.χ. λέξεις) σε κατηγορίες (π.χ. parts of speech) με βάση ένα σύνολο χαρακτηριστικών. Η είσοδος μπορεί να μετασχηματιστεί μαθηματικά χρησιμοποιώντας μια «kernel function» για να επιτρέψει τον γραμμικό διαχωρισμό των σημείων δεδομένων από διαφορετικές κατηγορίες. Δηλαδή, στην απλούστερη περίπτωση δύο χαρακτηριστικών, μια ευθεία γραμμή θα τα διαχωρίζει σε ένα X-Y επίπεδο: στη γενική περίπτωση N-χαρακτηριστικών, ο διαχωριστής θα είναι ένα (N-1) υπερ-επίπεδο. Η συνήθης kernel function που χρησιμοποιείται είναι Gaussian (η βάση της «κανονικής κατανομής» στις στατιστικές). Η διαδικασία διαχωρισμού επιλέγει ένα υποσύνολο των δεδομένων εκπαίδευσης τα «διανύσματα υποστήριξης» - σημεία δεδομένων

πλησιέστερα προς το βέλτιστο σημείο διαχωρισμού, που διαφοροποιεί καλύτερα τις κατηγορίες. Το διαχωριστικό υπερ-επίπεδο μεγιστοποιεί την απόσταση για να υποστηρίζει διανύσματα από κάθε κατηγορία (βλ. Σχήμα 1).



Εικόνα 5 Support vector machines

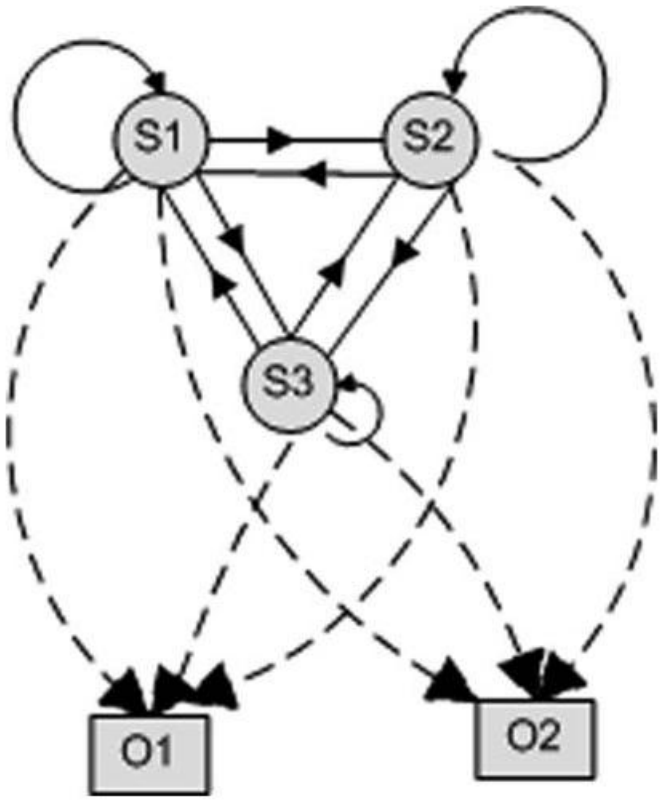
2.5.1.2 Hidden Markov models (HMMs)

Ένα HMM είναι ένα σύστημα όπου μια μεταβλητή μπορεί να μεταπτώσει (με διάφορες πιθανότητες) μεταξύ πολλών καταστάσεων, δημιουργώντας ένα από τα πολλά πιθανά σύμβολα εξόδου με κάθε μεταβολή της (επίσης με διάφορες πιθανότητες). Τα σύνολα πιθανών καταστάσεων και τα μοναδικά σύμβολα μπορεί να είναι μεγάλα αλλά πεπερασμένα και γνωστά (βλ. Σχήμα 2). Μπορούμε να παρατηρήσουμε τις εξόδους, αλλά τα εσωτερικά του συστήματος (δηλαδή οι πιθανότητες μεταβολής του κράτους και οι πιθανότητες εξόδου) είναι «κρυμμένες». Τα προβλήματα που πρέπει να λυθούν είναι:

Inference: δεδομένης μιας συγκεκριμένης ακολουθίας συμβόλων εξόδου, υπολογίζονται οι πιθανότητες μίας ή περισσοτέρων ακολουθιών υποψηφίων μεταβλητών.

• Αντιστοίχιση μοτίβων: βρίσκει την αλληλουχία του διακόπτη κατάστασης που είναι πιθανότερο να δημιουργήσει μια συγκεκριμένη ακολουθία συμβόλων εξόδου.

• Εκπαίδευση: δίδονται παραδείγματα των δεδομένων αλληλουχίας εξόδου-συμβόλου (training), υπολογίζονται οι πιθανότητες κατάστασης / μεταγωγής / εξόδου (δηλ. Εσωτερικά συστήματα) που ταιριάζουν καλύτερα σε αυτά τα δεδομένα.



Εικόνα 6 Hidden Markov models

Το S1 και S2 είναι συλλογιστική Naive Bayesian επεκτεινόμενη σε ακολουθίες, επομένως, τα HMMs χρησιμοποιούν ένα μοντέλο γενίκευσης. Για την επίλυση αυτών των προβλημάτων, ένα HMM χρησιμοποιεί δύο απλουστευτικές υποθέσεις (που είναι αληθινές για πολλά φαινόμενα πραγματικής ζωής):

1. Η πιθανότητα αλλαγής σε νέα κατάσταση (ή επιστροφή στην ίδια κατάσταση) εξαρτάται από τις προηγούμενες καταστάσεις N. Στην απλούστερη περίπτωση «πρώτης τάξης» (N = 1), αυτή η πιθανότητα καθορίζεται μόνο από την τρέχουσα κατάσταση. (Τα HMM πρώτης τάξης είναι επομένως χρήσιμα για να μοντελοποιήσουν συμβάντα των οποίων η πιθανότητα εξαρτάται από το τι συνέβη τελευταία).
2. Η πιθανότητα δημιουργίας μιας συγκεκριμένης εξόδου σε μια συγκεκριμένη κατάσταση εξαρτάται μόνο από αυτήν την κατάσταση

Αυτές οι υποθέσεις επιτρέπουν την υπολογισμό της πιθανότητας μιας αλληλουχίας μεταγωγής κατάστασης (και μιας αντίστοιχης ακολουθίας παρατηρούμενης εξόδου) με απλό πολλαπλασιασμό των ατομικών πιθανοτήτων. Υπάρχουν αρκετοί αλγόριθμοι για την επίλυση αυτών των προβλημάτων. Αποδοτικότερος όλων ο αλγόριθμος Viterbi, ο οποίος αντιμετωπίζει το πρόβλημα Β, βρίσκει εφαρμογές στην επεξεργασία σήματος, για παράδειγμα σε τεχνολογίες κινητών επικοινωνιών.

Θεωρητικά, τα HMM θα μπορούσαν να επεκταθούν σε ένα πολυπαραγοντικό σενάριο, αλλά το πρόβλημα της εκπαίδευσης μπορεί να τα καταστήσει ασύμφορα. Στην πράξη, οι εφαρμογές πολλαπλών μεταβλητών των HMMs (π.χ. NER68) χρησιμοποιούν απλές, εικονικές μεταβλητές που είναι μοναδικά προσδιορισμένες στη σύνθεση των υφιστάμενων απόλυτων μεταβλητών: οι προσεγγίσεις αυτές απαιτούν πολύ περισσότερα δεδομένα εκπαίδευσης.

Τα HMMs χρησιμοποιούνται ευρέως για την αναγνώριση ομιλίας, όπου η κυματομορφή μιας ομιλούμενης λέξης (η ακολουθία εξόδου) ταιριάζει με την ακολουθία των μεμονωμένων φωνημάτων τις «καταστάσεις» που πιθανότατα την παρήγαγαν. (Frederick Jelinek, ένας υπέρμαχος της στατιστικής NLP, ο οποίος πρωτοστάτησε στα HMMs στην ομάδα αναγνώρισης ομιλίας της IBM, ανέφερε πως αναρωτιέται το γιατί «κάθε φορά που ένας γλωσσολόγος εγκαταλείπει την ομάδα μου, βελτιώνεται η απόδοση του αναγνώστη ομιλίας») Τα HMMs αντιμετωπίζουν επίσης αρκετά προβλήματα βιοπληροφορικής, όπως στην ευθυγράμμιση πολλαπλών αλληλουχιών και την πρόβλεψη γονιδίων.

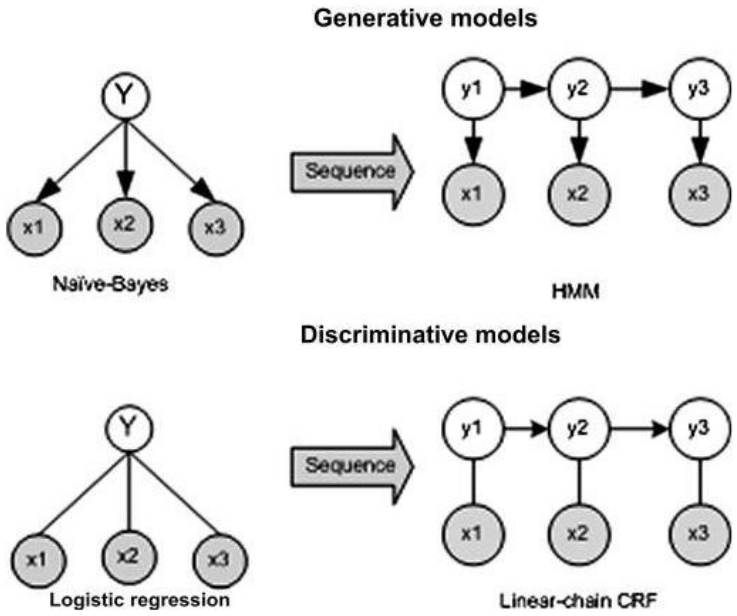
Τα εμπορικά συστήματα βασισμένα σε HMM για μετατροπή ομιλίας-προς-κείμενο είναι τώρα αρκετά ισχυρά ώστε να έχουν ουσιαστικά εκμηδενίσει τις ακαδημαϊκές ερευνητικές προσπάθειες, με συστήματα υπαγόρευσης για εξειδικευμένους τομείς - π.χ. ακτινολογία και παθολογία - παρέχοντας δομημένη εισαγωγή δεδομένων. Η αναγνώριση φράσεων είναι παράδοξα πιο αξιόπιστη για πολυσυλλάβους ιατρικούς όρους απ' ότι για τα συνηθισμένα αγγλικά: λίγες ακολουθίες λέξεων ακούγονται σαν «angina pectoris», ενώ τα κοινά αγγλικά έχουν πολλά ομόφωνα όπως π.χ. (two/too/to).

2.5.1.3 Conditional random fields (CRFs)

Τα CRFs είναι μια οικογένεια διακριτικών μοντέλων που προτάθηκαν αρχικά από τους Lafferty. Οι συνηθέστερες CRF μοιάζουν με HMMs στο ότι η επόμενη κατάσταση εξαρτάται από την τρέχουσα κατάσταση (εξ ου και η «εξάρτηση γραμμικής αλυσίδα»).

Τα CRF γενικεύουν την λογιστική παλινδρόμηση στα διαδοχικά δεδομένα με τον ίδιο τρόπο που τα HMM γενικεύουν τη Naive Bayes (βλ. Σχήμα). Τα CRF χρησιμοποιούνται για την πρόβλεψη των μεταβλητών κατάστασης («Ys») με βάση τις παρατηρούμενες μεταβλητές («Xs»). Για παράδειγμα, όταν εφαρμόζονται στο NER, οι μεταβλητές κατάστασης είναι οι κατηγορίες των ονομαζόμενων οντοτήτων NE: θέλουμε να προβλέψουμε μια ακολουθία

κατηγοριών ονομαζόμενων οντοτήτων μέσα σε ένα πέρασμα. Οι παρατηρούμενες μεταβλητές μπορεί να είναι η ίδια η λέξη, τα προθέματα ή επιθήματα, κεφαλαία γράμματα, ενσωματωμένοι αριθμοί, συλλαβισμός κ.ο.κ. Το παράδειγμα της CRF ταιριάζει με το NER: για παράδειγμα, εάν η προηγούμενη οντότητα είναι "Χαρακτηρισμός" (π.χ. "κος / κα."), Η επόμενη οντότητα πρέπει να είναι άτομο.



Εικόνα 7 Conditional random fields

Τα CRF είναι καλύτερα προσαρμοσμένα στα διαδοχικά δεδομένα πολλών μεταβλητών από τα HMMs: ενώ απαιτεί περισσότερα παραδείγματα, για την εκπαίδευση, από ένα απλό HMM, εξακολουθεί να είναι αποδοτικότερο.

2.6 Πεδία εφαρμογής μεθόδων NLP

2.6.1 Μηχανική μετάφραση

Καθώς το μεγαλύτερο μέρος του κόσμου είναι συνδεδεμένο στο διαδίκτυο, η δυνατότητα πρόσβασης σε δεδομένα είναι διαθέσιμη σε όλους. Σημαντική πρόκληση για την πρόσβαση στα δεδομένα είναι το γλωσσικό εμπόδιο. Υπάρχουν πολλές γλώσσες με διαφορετική δομή και γραμματική. Η μηχανική μετάφραση μεταφράζει γενικά φράσεις από τη μια γλώσσα στην άλλη με τη βοήθεια ενός στατιστικού μηχανισμού όπως το Google Translate. Η πρόκληση με τις τεχνολογίες μηχανικής μετάφρασης δεν είναι η απολυτή μετάφραση των λέξεων, αλλά η διατήρηση των εννοιών των προτάσεων μαζί με τη γραμματική και τις χρονικές στιγμές. Η

στατιστική μηχανική μάθηση συγκεντρώνει όσα δεδομένα μπορεί να διαπιστώσει ότι είναι σχετικά μεταξύ δύο γλωσσών και αναλύει τα δεδομένα τους για να βρει την πιθανότητα ότι κάτι στη Γλώσσα Α αντιστοιχεί σε κάτι στη Γλώσσα Β.

Όσον αφορά την Google, τον Σεπτέμβριο του 2016, η εταιρία παρουσίασε ένα νέο σύστημα μηχανικής μετάφρασης βασισμένο σε τεχνητά νευρωνικά δίκτυα και βαθιά μηχανική μάθηση. Τα τελευταία χρόνια έχουν προταθεί διάφορες μέθοδοι για την αυτόματη αξιολόγηση της ποιότητας της μηχανικής μετάφρασης, συγκρίνοντας τις υποδείξεις για διορθώσεις στις μεταφράσεις με τις μεταφράσεις αναφοράς. Παραδείγματα τέτοιων μεθόδων είναι ο ρυθμός σφάλματος λέξεων, ο ρυθμός σφάλματος λόγου ανεξάρτητος από τη θέση (position-independent word error rate) [14], η ακρίβεια του παραγόμενων λέξεων. Όλα αυτά τα κριτήρια προσπαθούν να προσεγγίσουν την αξιολόγηση του ανθρώπου και συχνά επιτυγχάνουν έναν εκπληκτικό βαθμό συσχέτισης στην ανθρώπινη υποκειμενική αξιολόγηση σε ευφράδεια και επάρκεια. [15]

2.6.2 Κατηγοριοποίηση κειμένου

Τα συστήματα κατηγοριοποίησης εισάγουν μια μεγάλες ροές δεδομένων όπως επίσημα έγγραφα, εκθέσεις στρατιωτικών ατυχημάτων, δεδομένα αγοράς, νέα μηνύματα κ.λπ. και τα αναθέτουν σε προκαθορισμένες κατηγορίες ή πίνακες ευρετήρια. Για παράδειγμα, το σύστημα Construe του Ομίλου Carnegie, εισάγει τα άρθρα του Reuters εξοικονομώντας πολύ χρόνο κάνοντας το έργο του προσωπικό ή των ειδικών επαγγελματιών δεικτοδοτών. Ορισμένες εταιρείες χρησιμοποιούν συστήματα κατηγοριοποίησης για να ταξινομούν τις αναφορές προβλημάτων (Tickets) ή τα αιτήματα καταγγελιών και να τα δρομολογούν στα κατάλληλα γραφεία.

Μια άλλη εφαρμογή κατηγοριοποίησης κειμένου είναι τα φίλτρα ανεπιθύμητης αλληλογραφίας. Τα φίλτρα ανεπιθύμητης αλληλογραφίας καθίστανται σημαντικά ως η πρώτη γραμμή άμυνας έναντι των ανεπιθύμητων μηνυμάτων ηλεκτρονικού ταχυδρομείου. Ένα ψευδές αρνητικό και ψευδώς θετικό ζήτημα των φίλτρων ανεπιθύμητης αλληλογραφίας βρίσκεται στο επίκεντρο της τεχνολογίας των NLP, αλλά είναι πρόβλημα το οποίο μειώνεται μετά την εξαγωγή της έννοιας από το κείμενο. Μια λύση φιλτραρίσματος ηλεκτρονικού ταχυδρομείου χρησιμοποιεί ένα σύνολο πρωτοκόλλων για να καθορίσει ποιο από τα εισερχόμενα μηνύματα είναι spam και ποιο όχι. Υπάρχουν διαθέσιμα διάφορα είδη φίλτρων ανεπιθύμητης αλληλογραφίας.

- **Φίλτρα περιεχομένου**: Ελέγχουν το περιεχόμενο εντός του μηνύματος για να προσδιορίσουν αν πρόκειται για spam ή όχι.
- **Φίλτρα επικεφαλίδας**: Ανατρέχουν στην κεφαλίδα του μηνύματος που αναζητά ψεύτικες πληροφορίες.

- **Φίλτρα γενικής μαύρης λίστας**: Καταργεί όλα τα μηνύματα ηλεκτρονικού ταχυδρομείου από τους παραλήπτες που περιλαμβάνονται στη λίστα.
- **Βασικά φίλτρα:** Χρησιμοποιεί κριτήρια που ορίζονται από το χρήστη. Όπως η διακοπή μηνυμάτων από συγκεκριμένο άτομο ή η διακοπή της αλληλογραφίας, όταν σε αυτά συμπεριλαμβάνεται μια συγκεκριμένη λέξη.
- **Φίλτρα αδειών**: Απαιτείται ο καθένας να στέλνει ένα μήνυμα για να εγκριθεί εκ των προτέρων από τον παραλήπτη.
- **Φίλτρα απόκρισης σε πρόκλησης**: Απαιτεί ο καθένας να στείλει ένα μήνυμα και να εισαγάγει έναν κωδικό για να αποκτήσει την άδεια αποστολής μηνυμάτων ηλεκτρονικού ταχυδρομείου.

2.6.3 Φιλτράρισμα ανεπιθύμητης αλληλογραφία

Λειτουργεί με την κατηγοριοποίηση των κειμένων. Πρόσφατα εφαρμόστηκαν διάφορες τεχνικές εκμάθησης μηχανών για την κατηγοριοποίηση κειμένων ή για το Φιλτράρισμα Anti-Spam, Naïve Bayes, Εκμάθηση με βάση τη μνήμη, Μοντέλο μέγιστης εντροπίας.

Μερικές φορές συνδυάζοντας διαφορετικούς τρόπους εκμάθησης [16]. Η χρήση αυτών των προσεγγίσεων είναι καλύτερη, καθώς ο ταξινομητής μαθαίνει από τα δεδομένα εκπαίδευσης και όχι μέσω της ανθρώπινης παρέμβασης. Ο naïve bayes προτιμάται λόγω της απόδοσής έχοντας παράλληλα αξιοσημείωτη απλότητα.

Στο φιλτράρισμα ανεπιθύμητης αλληλογραφίας έχουν χρησιμοποιηθεί δύο τύποι μοντέλων. Και τα δύο μοντέλα υποθέτουν ότι υπάρχει ένα σταθερό λεξιλόγιο. Στο πρώτο μοντέλο δημιουργείται ένα έγγραφο, επιλέγοντας πρώτα ένα υποσύνολο λεξιλογίου και στη συνέχεια χρησιμοποιώντας τις επιλεγμένες λέξεις πολλαπλές φορές, τουλάχιστον μία φορά ανεξάρτητα από τη σειρά. Αυτό ονομάζεται μοντέλο πολλαπλών ποικιλιών Bernoulli. Παίρνει τις πληροφορίες για ποιες λέξεις χρησιμοποιούνται σε ένα έγγραφο ανεξάρτητα από τον αριθμό των λέξεων και της σειράς.

Στο δεύτερο μοντέλο, δημιουργείται ένα έγγραφο επιλέγοντας ένα σύνολο λέξεων και ταξινομώντας τα με οποιαδήποτε σειρά. Αυτό το μοντέλο ονομάζεται πολυωνυμικό μοντέλο, εκτός από το μοντέλο Multi-variate Bernoulli, συλλαμβάνει επίσης πληροφορίες σχετικά με το πόσες φορές χρησιμοποιείται μια λέξη σε ένα έγγραφο. Οι περισσότερες προσεγγίσεις κατηγοριοποίησης κειμένων για το φιλτράρισμα ηλεκτρονικού ταχυδρομείου, για την ελαχιστοποίηση της ανεπιθύμητης αλληλογραφίας, έχουν χρησιμοποιήσει πολυπύρηνο μοντέλο Bernoulli.

2.6.4 Εξαγωγή πληροφοριών

Η εξαγωγή πληροφοριών (information extraction IE) αφορά τον προσδιορισμό των φράσεων ενδιαφέροντος των δεδομένων μορφής κειμένου. Για πολλές εφαρμογές, η εξαγωγή οντοτήτων όπως τα ονόματα, οι τόποι, τα γεγονότα, οι ημερομηνίες, οι χρόνοι και οι τιμές είναι ένας ισχυρός τρόπος να συνοψίσουμε τις πληροφορίες που σχετίζονται με τις ανάγκες ενός χρήστη. Στην περίπτωση μηχανών αναζήτησης συγκεκριμένου πεδίου, ο αυτόματος προσδιορισμός σημαντικών πληροφοριών μπορεί να αυξήσει την ακρίβεια και την αποτελεσματικότητα μιας κατευθυνόμενης αναζήτησης. Γίνεται χρήση κρυφών μοντέλων Markov (HMMs) για την εξαγωγή των σχετικών πεδίων των ερευνητικών εργασιών. Αυτά τα εξαγόμενά τμήματα κειμένου χρησιμοποιούνται για την αναζήτηση σε συγκεκριμένα πεδία και για την αποτελεσματική παρουσίαση των αποτελεσμάτων αναζήτησης και για την αντιστοίχιση των αναφορών σε έγγραφα. Παραδείγματος χάριν, παρατηρώντας τις αναδυόμενες διαφημίσεις σε οποιονδήποτε ιστότοπο που δείχνει τα πρόσφατα στοιχεία που εμφανίζονταν σε ένα ηλεκτρονικό κατάστημα με εκπτώσεις.

Στην Ανάκτηση Πληροφοριών έχουν χρησιμοποιηθεί δύο τύποι μοντέλων [17]. Και οι δύο περιπτώσεις υποθέτουν ότι υπάρχει ένα σταθερό λεξιλόγιο. Στο πρώτο μοντέλο δημιουργείται ένα έγγραφο επιλέγοντας πρώτα ένα υποσύνολο του λεξικού και στη συνέχεια χρησιμοποιώντας τις επιλεγμένες λέξεις οσεσδήποτε φορές, τουλάχιστον μία φορά χωρίς εντολή. Αυτό ονομάζεται μοντέλο πολλαπλών ποικιλιών Bernoulli. Παίρνει τις πληροφορίες για το ποιες λέξεις χρησιμοποιούνται σε ένα έγγραφο ανεξάρτητα από τον αριθμό των λέξεων και της σειράς. Στο δεύτερο μοντέλο, δημιουργείται ένα έγγραφο επιλέγοντας ένα σύνολο λέξεων και ταξινομώντας τα με οποιαδήποτε σειρά.. Το μοντέλο αυτό ονομάζεται πολυωνυμικό μοντέλο. Το μοντέλο πολλαπλών ποικιλιών Bernoulli, περιλαμβάνει επίσης πληροφορίες σχετικά με το πόσες φορές μια λέξη χρησιμοποιείται σε ένα έγγραφο

Η ανακάλυψη της γνώσης αποτελεί ένα πολύ σημαντικό τομέα έρευνας τα τελευταία χρόνια. Η έρευνα σχετικά με την ανακάλυψη γνώσης χρησιμοποιεί μια ποικιλία τεχνικών για την εξαγωγή χρήσιμων πληροφοριών από έγγραφα.

Μια τεχνική είναι η προσθήκη ετικετών (tags) στα μέρη ομιλίας, η ανάλυση Chunking ή Shadow, τα Stop-words (Λέξεις-κλειδιά που χρησιμοποιούνται και πρέπει να καταργηθούν πριν από την επεξεργασία εγγράφων).

Μια σημαντική τεχνική είναι το Stemming στο οποίο γίνεται η καταγραφή των λέξεων σε μία βάση-ρίζα, η τεχνική βασίζεται σε δύο μεθόδους, η μια στη χρήση λεξικών και η δεύτερη στην τεχνική Porter (Porter, 1980). Η πρώτη έχει υψηλότερη ακρίβεια αλλά και μεγάλο κόστος υλοποίησης, ενώ η δεύτερη έχει χαμηλότερο κόστος υλοποίησης και είναι συνήθως ανεπαρκής για IR).

Word Sense Αποσαφήνιση λέξεων (Διαισθητική τεχνική του Word είναι η κατανόηση της σωστής έννοιας μιας λέξης) Όταν χρησιμοποιούνται για την ανάκτηση πληροφοριών, οι όροι αντικαθίστανται από τις καταλληλότερες τους οι οποίες υπάρχουν στο διάνυσμα του εγγράφου.)

Οι εξαγόμενες πληροφορίες μπορούν να εφαρμοστούν για διάφορους σκοπούς, όπως για παράδειγμα για την προετοιμασία μιας περίληψης, για τη δημιουργία βάσεων δεδομένων, την αναγνώριση λέξεων-κλειδιών, την ταξινόμηση στοιχείων κειμένου σύμφωνα με ορισμένες προκαθορισμένες κατηγορίες κλπ. Για παράδειγμα το CONSTRUE αναπτύχθηκε για το [18]. Έχει προταθεί ότι πολλά συστήματα εξόρυξης πληροφορίας (IE) μπορούν να εξαγάγουν με επιτυχία όρους από έγγραφα, αλλά η εξαγωγή σχέσεων μεταξύ των όρων εξακολουθεί να αποτελεί δύσκολο ζήτημα. Το PROMETHEE είναι ένα σύστημα που εξάγει λεξικό-συντακτικά μοτίβα σε σχέση με μια συγκεκριμένη εννοιολογική σχέση. Τα συστήματα εξόρυξης πληροφορίας IE θα πρέπει να λειτουργούν σε πολλά επίπεδα, από την αναγνώριση λέξεων έως την ανάλυση του λόγου σε επίπεδο πλήρους εγγράφου. Μια εφαρμογή της προσέγγισης για την ανάλυση ενός πραγματικού φυσικού γλωσσικού σώματος αποτελείται από απαντήσεις σε ερωτηματολόγια ανοιχτού τύπου στον τομέα της διαφήμισης .

Υπάρχει ένα σύστημα το οποίο ονομάζεται MITA (Intelligent Text Analyzer της Metlife) το οποίο εξάγει πληροφορίες από εφαρμογές ασφάλισης ζωής, το οποίο πρότεινε ένα γενικό πλαίσιο για την εξόρυξη κειμένου που χρησιμοποιεί πραγματιστικές αναλύσεις κειμένου σε πραγματολογικό επίπεδο και επίπεδο ομιλίας.

2.6.5 Σύνοψη

Η υπερφόρτωση των πληροφοριών αποτελεί μια πραγματικότητα στην σημερινή ψηφιακή εποχή και η ευκολία πρόσβασης στην γνώση και τις πληροφορίες υπερβαίνει την ικανότητά μας να τις κατανοήσουμε. Αυτή η τάση δεν επιβραδύνεται, επομένως απαιτείται ιδιαίτερη ικανότητα να συνοψίζουμε τα δεδομένα διατηρώντας παράλληλα άθικτη την έννοια τους. Αυτό είναι σημαντικό όχι μόνο για να μας επιτρέψει να αναγνωρίσουμε και να κατανοήσουμε τις σημαντικές πληροφορίες για ένα μεγάλο σύνολο δεδομένων, αλλά και για να κατανοήσουμε τις βαθύτερες συναισθηματικές έννοιες. Για παράδειγμα, μια εταιρεία καθορίζει το γενικό συναίσθημα στα κοινωνικά μέσα δικτύωσης, προσαρμόζοντας την καμπάνια των προϊόντων τους. Αυτή η περίπτωση χρήσης είναι πολύτιμο στοιχείο για το τμήμα μάρκετινγκ.

Οι τύποι περιλήψεων κειμένων εξαρτώνται από των αριθμό των εγγράφων και οι δύο σημαντικές κατηγορίες είναι η συνοπτική παρουσίαση ενός ενιαίου εγγράφου και η συνοπτική παρουσίαση πολλών εγγράφων [19][20]. Οι περιλήψεις μπορούν επίσης να είναι δύο τύπων: γενικές ή να εξαρτώνται από το ερώτημα. Η διαδικασία της συνοπτικής παρουσίασης μπορεί να είναι είτε υπό επίβλεψη είτε χωρίς επίβλεψη. Δεδομένα εκπαίδευσης απαιτούνται σε ένα

εποπτευόμενο σύστημα για την επιλογή σχετικού υλικού από τα έγγραφα. Απαιτείται μεγάλη ποσότητα υποσημειώσεων δεδομένων για τις τεχνικές μάθησης. Ορισμένες σημαντικές τεχνικές παρατίθενται παρακάτω:

- Το μοντέλο θεματικής βάσης Bayesian Sentence (BSTM) χρησιμοποιεί τόσο προτάσεις όρων όσο και ομαδοποιημένες ενώσεις εγγράφων για την περίληψη πολλαπλών εγγράφων.
- Η παραγοντοποίηση με δεδομένες βάσεις (FGB) είναι ένα γλωσσικό μοντέλο όπου οι βάσεις των προτάσεων είναι οι δοσμένες βάσεις και χρησιμοποιεί πίνακες όρων εγγράφου και πρότασης. Αυτή η προσέγγιση συγκεντρώνει και συνοψίζει ταυτόχρονα τα έγγραφα.
- Η θεματική συνοπτική παρουσίαση (Topic Aspect-Oriented Summarization TAOS) βασίζεται σε θεματικούς παράγοντες. Αυτοί οι θεματικοί παράγοντες είναι διάφορα χαρακτηριστικά που περιγράφουν θέματα όπως οι λέξεις κεφαλαίου που χρησιμοποιούνται για να αντιπροσωπεύουν την οντότητα. Διάφορα θέματα μπορούν να έχουν διάφορες πτυχές και διάφορες προτιμήσεις των χαρακτηριστικών χρησιμοποιούνται για να αντιπροσωπεύουν διάφορες πτυχές.

2.6.6 Σύστηματα διαλόγου

Ίσως η πλέον επιθυμητή εφαρμογή του μέλλοντος, στα συστήματα που προβλέπουν οι μεγάλοι πάροχοι υπηρεσιών τελικών χρηστών, είναι τα συστήματα διαλόγου, που επικεντρώνονται σε πολύ συγκεκριμένες εφαρμογές. Τα συστήματα διαλόγου, με την χρήση όλων των επίπεδων της επεξεργασίας γλώσσας, προσφέρουν δυνατότητες για πλήρως αυτοματοποιημένα συστήματα διαλόγου, είτε σε επίπεδο κείμενου είτε σε επίπεδο φωνής. Αυτό θα μπορούσε να οδηγήσει στην παραγωγή συστημάτων που θα επιτρέπουν στα ρομπότ να αλληλοεπιδρούν με τους ανθρώπους χρησιμοποιώντας φυσική γλώσσα. Παραδείγματα όπως ο βοηθός της Google, τα Windows Cortana, το Siri της Apple και το Alexa του Amazon αποτελούν λογισμικό που κάνουν χρήση των συστημάτων διαλόγου.

2.6.7 Ιατρική

Το NLP εφαρμόζεται επίσης στον τομέα της ιατρικής. Το Πρόγραμμα Γλωσσολογικής - Επεξεργασίας Ιατρικής Γλώσσας (The Linguistic String Project-Medical Language Processor LSP-MLP) είναι ένα μεγάλο πρόγραμμα NLP στον τομέα της ιατρικής. Το LSP-MLP βοηθάει τους ιατρούς να εξάγουν και να συνοψίζουν τις πληροφορίες για τυχόν σημεία ή συμπτώματα, δεδομένα δοσολογίας και απόκρισης φαρμάκων με στόχο τον εντοπισμό πιθανών παρενεργειών οποιουδήποτε φαρμάκου, καθώς και για την επισήμανση σημαντικών

δεδομένων . Η Εθνική Βιβλιοθήκη Ιατρικής αναπτύσσει ένα ειδικό σύστημα, το οποίο αναμένεται να λειτουργήσει ως εργαλείο εξαγωγής πληροφοριών για τις βάσεις γνώσεων της βιοϊατρικής. Το λεξικό δημιουργήθηκε χρησιμοποιώντας το MeSH (Medical Subject Headings), το οποίο είναι ένα εικονογραφημένο ιατρικό λεξικό της Dorland και χρησιμοποιεί γενικά αγγλικά λεξικά.

Το Center d'Informatique Hospitaliere του Hopital Cantonal de Geneve εργάζεται σε ένα ηλεκτρονικό περιβάλλον αρχειοθέτησης με χαρακτηριστικά NLP. Στην πρώτη φάση, τα αρχεία ασθενών αρχειοθετήθηκαν. Σε μεταγενέστερο στάδιο, το LSP-MLP έχει προσαρμοστεί για τα γαλλικά και, τέλος, έχει αναπτυχθεί ένα κατάλληλο σύστημα NLP που ονομάζεται RECIT με μια μέθοδο που ονομάζεται Proximity Processing. Στόχος του ήταν να αναπτυχθεί ένα ισχυρό πολύγλωσσο σύστημα ικανό να αναλύσει / κατανοήσει τις ιατρικές προτάσεις και να διατηρήσει τη γνώση του ελεύθερου κειμένου σε μια αναπαράσταση της γνώσης ανεξάρτητη από τη γλώσσα.

Το πανεπιστήμιο της Κολούμπια της Νέας Υόρκης έχει αναπτύξει ένα σύστημα NLP που ονομάζεται MEDLEE (σύστημα εξαγωγής και κωδικοποίησης ιατρικής γλώσσας) το οποίο προσδιορίζει κλινικές πληροφορίες σε αναφορές αφηγήσεων και μετατρέπει τις πληροφορίες κειμένου σε δομημένη αναπαράσταση.

3

Ανάκτηση πληροφορίας, μηχανές αναζήτησης

3.1 Εισαγωγή

Έχοντας εξετάσει τα χαρακτηριστικά και τα κίνητρα των συστημάτων NLP, επικεντρωνόμαστε στο πρόβλημα της πρόσβασης στις πληροφορίες και ειδικότερα στην ανάκτηση πληροφορίας και στις ταξινομίες.

Σε αυτό το κεφάλαιο, εξετάζουμε συνοπτικά τις επικρατούσες προσεγγίσεις στην ανάκτηση πληροφοριών. Αρχικά εξετάζουμε τις κυρίαρχες προσεγγίσεις για την αναζήτηση κειμένου: Την καθορισμένη ανάκτηση (set retrieval) και την ανάκτηση με κατάταξη (ranked retrieval). Στη συνέχεια, μελετούμε την πλοήγηση βάση καταλόγου: μια προσέγγιση η οποία, αν και δεν είναι συγκεκριμένη για τις συλλογές κειμένων, έχει συχνά εφαρμοστεί σε αυτές.

Σε αυτό το σύντομο κεφάλαιο δεν επιχειρούμε την εξαντλητική μελέτη της ανάκτησης πληροφοριών - ένα θέμα που καλύπτει τουλάχιστον έξι δεκαετίες, και ξεκινάει από το όραμα της memex του Vannevar Bush έως τις μηχανές αναζήτησης ιστού, όπως αυτές που παρέχονται από το Google, το Yahoo και τη Microsoft, αποτελώντας πλέον αναπόσπαστο κομμάτι της ζωής μας σήμερα. Εστιάζουμε στις βασικές μεθόδους ανάκτησης εγγράφων, αγνοώντας περιοχές όπως τα πολυμέσα και την αναζήτηση στα κοινωνικά μέσα δικτύωσης.

3.2 Σχετικότητα (relevance)

Βασική μέριμνα της ανάκτησης πληροφοριών είναι να βοηθήσει τους χρήστες να ανακτήσουν έγγραφα που σχετίζονται με τις ανάγκες πληροφόρησης τους. Ωστόσο, η έννοια της συνάφειας είναι δύσκολο να καθοριστεί. Όπως έγραψε ο William Goffman το 1964 [21]: Η συνάφεια ορίζεται ως ένα μέτρο πληροφορίας που μεταφέρεται από ένα έγγραφο σε σχέση

με ένα ερώτημα. Η σχέση μεταξύ του εγγράφου και του ερωτήματος, αν και απαραίτητη, δεν επαρκεί πάντα για τον προσδιορισμό της συνάφειας

Ο Stefano Mizzaro και ο Tefko Saracevic έχουν γράψει μελέτες, καταγράφοντας και τεκμηριώνοντας την εξέλιξη αυτής της ιδέας μεταξύ των επιστημόνων οι οποίοι εργάσθηκαν σε βιβλιοθήκες και των ερευνητών ανάκτησης πληροφοριών που συχνά διαφώνησαν σχετικά με τον τρόπο μέτρησης της [22, 23]. Οι πρώτοι τείνουν να υιοθετούν μια γνωστική προσέγγιση με γνώμονα τον χρήστη, ενώ οι τελευταίοι υιοθετούν μια προσέγγιση που βασίζεται σε συγκριτικά κριτήρια. Αυτή η φιλοσοφική διαφορά οδηγεί σε διαφορετικές προσεγγίσεις αξιολόγησης. Με τους μεν πρώτους να ευνοούν τις μελέτες χρηστών και τους μεν δεύτερους να ευνοούν τη χρήση συλλογών εκπαίδευσης, ιδιαίτερα εκείνων που διατηρεί το Text Retrieval Conference (TREC) [24].

Δυστυχώς, η περίπτωση χρήσης της μεθόδου TREC για τη μέτρηση της συνάφειας δεν αποδείχθηκε αποτελεσματική για τα συστήματα διαδραστικής ανάκτησης πληροφοριών και οι μελέτες χρηστών μπορεί να είναι απαγορευτικά δαπανηρές. Ως εκ τούτου, οι ερευνητές ανάκτησης πληροφοριών δέχονται έναν ορισμό της συνάφειας ως ένα μέτρο που μεταφέρεται εξ ολοκλήρου από ένα έγγραφο σε σχέση με ένα ερώτημα (Goffman). Οι επιστήμονες οι οποίοι εργάζονται στην αναζήτηση πληροφοριών σε έγραφα βιβλιοθηκών τηρούν μια προσέγγιση με επίκεντρο τον χρήστη, η οποία συνήθως μετρά την αποτελεσματικότητα των συστημάτων υποστήριξης που αναζητούν πληροφορίες, μέσω μελετών με συμμετοχή χρηστών, στο επίπεδο της μεθόδου αναζήτησης και όχι στο επίπεδο του ερωτήματος.

3.3 Καθορισμένη ανάκτηση (set retrieval)

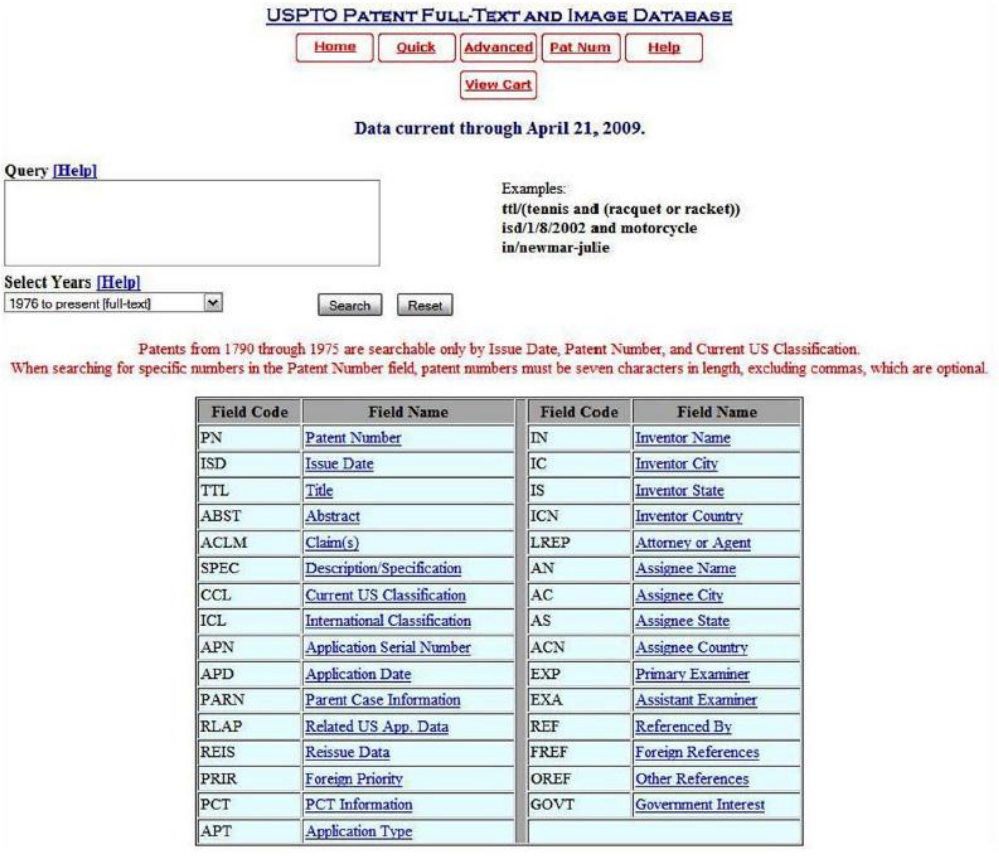
Αν και η αναζήτηση είναι πανταχού παρούσα σήμερα, οι πρώτες μηχανές αναζήτησης ή πιο τυπικά, τα πρώιμα συστήματα ανάκτησης πληροφοριών - λειτουργούσαν πολύ διαφορετικά από τους σύγχρονους ομολόγους τους. Σε αντίθεση με τις περισσότερες σύγχρονες μηχανές αναζήτησης, τα πρώτα συστήματα ανάκτησης πληροφοριών, χρησιμοποίησαν ένα μοντέλο ανάκτησης [25]. Τα συστήματα καθορισμένης ανάκτησης επιστρέφουν αποτελέσματα, τα οποία είναι σύνολα εγγράφων παρά και όχι ταξινομημένες ακολουθίες, με αποτέλεσμα ουσιαστικά να μην υπάρχει η έννοια της κατάταξης με βάση την σχετικότητα.

Αυτό το μοντέλο είναι επίσης γνωστό ως μοντέλο ανάκτησης Boolean (Boolean retrieval), επειδή τα συστήματα ανάκτησης, επιτρέπουν στους χρήστες να καθορίζουν τις εκφράσεις των ερωτημάτων τους χρησιμοποιώντας λειτουργίες Boolean (AND, OR, NOT). Πολλά από αυτά τα συστήματα έχουν επεκτείνει τη σύνταξη Boolean για να συμπεριλάβουν πρόσθετους δείκτες ώστε να καθορίσουν τη σειρά των όρων ή την εγγύτητα των όρων μέσα σε ένα έγγραφο. Ορισμένα συστήματα επιτρέπουν επίσης στους χρήστες να καθορίζουν σε ποιο τμήμα σε ένα έγγραφο προέκυψε ένας όρος (π.χ. στο πεδίο τίτλου ή σε αυτό του πεδίου συγγραφέα), ένα

θέμα στο οποίο θα επιστρέψουμε όταν συζητάμε την παραμετρική και την πολύ- παραμετρική αναζήτηση.

Το εικόνα 8 δείχνει ένα παράδειγμα αναζήτησης τύπου Boolean, το οποίο χρησιμοποιεί δεδομένα από τη προηγμένη σελίδα αναζήτησης για τη βάση δεδομένων των διπλωμάτων ευρεσιτεχνίας των ΗΠΑ. Οι χρήστες μπορούν να καθορίσουν ερωτήματα σε μια γλώσσα η οποία περιλαμβάνει τυπικούς τελεστές Boolean και με τον τρόπο αυτό να περιορίζουν τα αποτελέσματα τους.

Παρά την ευελιξία αυτή, η δυνατότητα ανάκτησης πάσχει από μια θεμελιώδη αδυναμία. Καθώς οι χρήστες επιχειρούν να εκφράσουν τις ανάγκες πληροφοριών τους ως ερωτήματα Boolean, συχνά βρίσκονται αντιμέτωποι με μια επιλογή μεταξύ υψηλής ακρίβειας (Precision) και υψηλής ανάκλησης (Recall) αλλά δεν είναι σε θέση να επιτύχουν επαρκή ακρίβεια και ανάκληση. Πράγματι, η δυσκολία που αντιμετωπίζουν οι χρήστες στη διαμόρφωση των ερωτημάτων Boolean μπορεί να εξηγήσει τη σπανιότητα των ερωτημάτων αναζήτησης στο Web, παρόλο που οι περισσότερες μηχανές αναζήτησης Ιστού υποστηρίζουν Boolean τελεστές [26].



Field Code	Field Name	Field Code	Field Name
PN	Patent Number	IN	Inventor Name
ISD	Issue Date	IC	Inventor City
TTL	Title	IS	Inventor State
ABST	Abstract	ICN	Inventor Country
ACLM	Claim(s)	LREP	Attorney or Agent
SPEC	Description/Specification	AN	Assignee Name
CCL	Current US Classification	AC	Assignee City
ICL	International Classification	AS	Assignee State
APN	Application Serial Number	ACN	Assignee Country
APD	Application Date	EXP	Primary Examiner
PARN	Parent Case Information	EXA	Assistant Examiner
RLAP	Related US App. Data	REF	Referenced By
REIS	Reissue Data	FREF	Foreign References
PRIR	Foreign Priority	OREF	Other References
PCT	PCT Information	GOVT	Government Interest
APT	Application Type		

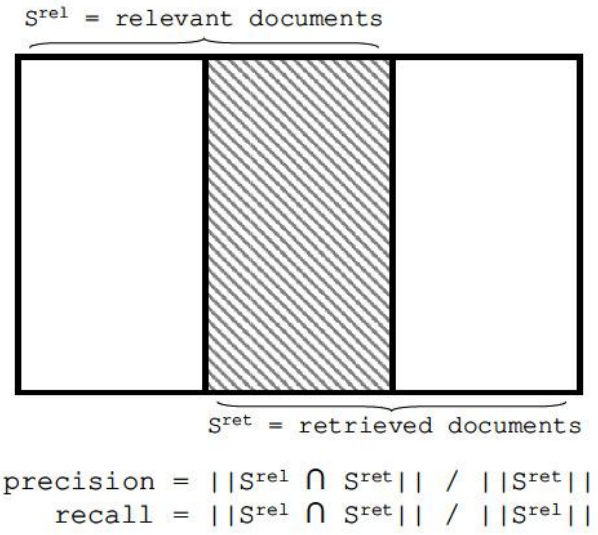
Εικόνα 8 Η διεπαφή Boolean αναζήτησης του Γραφείου Ευρεσιτεχνιών των ΗΠΑ

Ας ορίσουμε αυτούς τους όρους και στη συνέχεια να εξηγήσουμε το συμβιβασμό μεταξύ τους.

3.4 Ακρίβεια έναντι ανάκτησης

Στη βιβλιογραφία η οποία σχετίζεται με την ανάκτηση πληροφοριών, η ακρίβεια και η ανάκληση αποτελούν τα δύο πιο διαδεδομένα μέτρα για την ακρίβεια ανάκτησης και συχνά ονομάζεται απόδοση ανάκτησης (retrieval performance) στη βιβλιογραφία, αλλά προτιμούμε τον όρο accuracy επειδή πολλοί άνθρωποι χρησιμοποιούν τον "performance" για να χαρακτηρίσουν ταχύτητα ή υπολογιστική αποδοτικότητα.

Για ένα δεδομένο ερώτημα, η ακρίβεια είναι το κλάσμα των ανακτηθέντων εγγράφων που σχετίζονται με την ανάγκη πληροφόρησης που αντιπροσωπεύει το ερώτημα. Ανάκληση είναι το κλάσμα όλων των πιθανών σχετικών εγγράφων που ανακτώνται. Αν εξετάσουμε τον δικαστικό όρκο να «να πούμε την αλήθεια, ολόκληρη την αλήθεια και τίποτα άλλο από την αλήθεια», η ανάκληση μετράει το βαθμό στον οποίο ένα σύστημα ανάκτησης πληροφοριών λέει ολόκληρη την αλήθεια και η ακρίβεια μετράει τον βαθμό στον οποίο δεν λέει τίποτα παρά αλήθεια.



Εικόνα 9Ακρίβεια και ανάκληση στο μοντέλο της καθορισμένης ανάκτησης

Στην ιδανική περίπτωση, ένα σύστημα ανάκτησης θα επιτυγχάνει τόσο υψηλή ακρίβεια όσο και υψηλή ανάκληση. Όπως αποδεικνύεται, αυτό είναι δύσκολο να γίνει χρησιμοποιώντας μια τυπική διασύνδεση ανάκτησης τύπου Boolean. Ας χρησιμοποιήσουμε ένα συγκεκριμένο παράδειγμα για να ώστε να καταλάβουμε καλύτερα τη δυσκολία αυτή: φανταστείτε ότι ψάχνουμε για εργασία στην επιστήμη της βιβλιοθήκης σε μια συλλογή επιστημονικής λογοτεχνίας

Μπορούμε να επιτύχουμε υψηλή ακρίβεια ψάχνοντας για έργα που περιέχουν την ακριβή φράση " library science " - και μπορεί να επιτευχθεί σχεδόν 100% ακρίβεια θεωρώντας μόνο τα έργα που περιέχουν αυτή τη φράση στον τίτλο τους. Ο Stephen Robertson αναφέρεται σε τέτοια στοιχεία ως precision devices" [27]. Αν και ένας συνδυασμός συσκευών ακριβείας μπορεί να επιτύχει επαρκή βαθμό ακρίβειας, είναι πιθανό να οδηγήσει σε κακή ανάκληση. Για παράδειγμα, κανένα από τα έργα του Ranganathan δεν περιλαμβάνει τη φράση "science library" στον τίτλους τους. Μπορούμε να επιλέξουμε μια στρατηγική που έχει σαν προτεραιότητα την ανάκληση για παράδειγμα, επιστρέφοντας όλα τα έργα των οποίων το πλήρες κείμενο περιέχει είτε " library" είτε "science" ή τυχόν παραλλαγές αυτών των λέξεων (π.χ. librarian, scientist).

Θα μπορούσαμε να αυξήσουμε περαιτέρω την ανάκληση μέσω της επέκτασης της λεκτικής εγκυκλοπαίδειας ‘thesaurus’ (π.χ. repository, collection, technology). Μια τέτοια προσέγγιση θα μπορούσε αναμφισβήτητα να επιτύχει υψηλή ανάκληση, αλλά επηρεάζοντας αρνητικά σε μεγάλο βάρος την ακρίβεια (π.χ., λιγότερο από 10%).

Αν και η επέκταση που περιγράφεται μπορεί να φανεί τόσο ακραία ώστε να είναι παράλογη, η εύρεση μιας μέσης κατάστασης αποδεικνύεται δύσκολη. Οι επαγγελματίες βιβλιοθηκονόμοι είναι καλύτεροι από τους ερασιτέχνες που αναζητούν πληροφορίες, στην κατασκευή σύνθετων ερωτημάτων Boolean για τη βελτιστοποίηση της ισορροπίας μεταξύ ακριβείας και ανάκλησης. Παρ 'όλα αυτά, ακόμη και αυτοί πολλές φορές υπερεκτιμούν τις ικανότητες στην αποτελεσματική τη χρήση των Boolean τεχνικών.

Μια μελέτη του 1994 από την West Publishing Company (η οποία παρέχει την υπηρεσία πληροφοριών Westlaw σε νομικούς επαγγελματίες) κατέδειξε ότι ακόμη και οι έμπειροι αναζητητές πληροφοριών ήταν καλύτερα να εγκαταλείψουν την Boolean ανάκτηση υπέρ της απλής εισαγωγής λέξεων σε ένα πλαίσιο αναζήτησης ως ερώτημα ελεύθερου κειμένου βασιζόμενοι στη μηχανή αναζήτησης ώστε να επιστρέψει ό, τι θεώρησε ότι ήταν το καλύτερο αποτέλεσμα [28]. Φαίνεται πλέον ασφαλές να υποθέσουμε ότι μετά από αρκετά χρόνια μελέτης, οι προγραμματιστές μηχανών αναζήτησης έχουν αυξήσει την αποτελεσματικότητα των αλγορίθμων τους περισσότερο από ό, τι οι ειδικοί που αναζητούν πληροφορίες έχουν βελτιώσει την κατάρτιση τους στην κατάρτιση ερωτημάτων.

3.5 Αξιολόγηση στην εξόρυξη πληροφορίας

Με δεδομένο ένα κείμενο εισόδου ή μια συλλογή κειμένων, η αναμενόμενη έξοδος ενός συστήματος IE μπορεί να οριστεί με ακρίβεια. Αυτό διευκολύνει την αξιολόγηση διαφορετικών συστημάτων και προσεγγίσεων. Συγκεκριμένα, οι μετρήσεις precision και recall υιοθετήθηκαν από την ερευνητική κοινότητα του IR για το σκοπό αυτό. Τα στοιχεία τα οποία καταγράφονται είναι η αποτελεσματικότητα του συστήματος από την οπτική γωνία του χρήστη, δηλαδή ο

βαθμός στον οποίο το σύστημα παράγει όλη την κατάλληλη έξοδο (recall) και μόνο την κατάλληλη έξοδο (precision). Έτσι, το recall και το precision μπορούν να θεωρηθούν ως μέτρο πληρότητας και ορθότητας, αντίστοιχα. Για να τα ορίσουμε τυπικά, εάν θεωρήσουμε πως το κλειδί # δηλώνει τον συνολικό αριθμό των διαθέσιμων σχισμών οι οποίες αναμένουμε να χρησιμοποιηθούν σύμφωνα με το δοθέν λεξικό, το οποίο αντιπροσωπεύει το σημείο αναφοράς. Εάν τεθεί σαν #correct ή (#incorrect)) ο αριθμός των ορθός (Λανθασμένα) συμπληρωμένων σχισμών για την απόκριση του συστήματος. Μια υποδοχή θεωρούμε ότι έχει γεμίσει λανθασμένα είτε εάν δεν ευθυγραμμίζεται με μια υποδοχή βάση του προτύπου είτε αν έχει εκχωρηθεί μια μη έγκυρη τιμή.

$$precision = \frac{\#correct}{\#correct + \#incorrect} \qquad recall = \frac{\#correct}{\#key}$$

Προκειμένου να έχουμε μια πιο λεπτομερή εικόνα της απόδοσης των συστημάτων IE, το recall και το precision συχνά μετριούνται για κάθε τύπο slot χωριστά. [29]

Το f-μέτρο χρησιμοποιείται ως σταθμισμένος αρμονικός μέσος όρος ακρίβειας και ανάκλησης.

$$F = \frac{(\beta^2 + 1) \times precision \times recall}{(\beta^2 \times precision) + recall}$$

Στον παραπάνω ορισμό το β είναι μη αρνητική τιμή που χρησιμοποιείται για να προσαρμόσει τη σχετική στάθμιση (το β = 1.0 δίνει την ίδια έμφαση στην ανάκληση και την ακρίβεια, και στην περίπτωση μικρότερων τιμών δίνεται αυξημένο βάρος στην ακρίβεια).

3.6 Ανάκτηση με κατάταξη

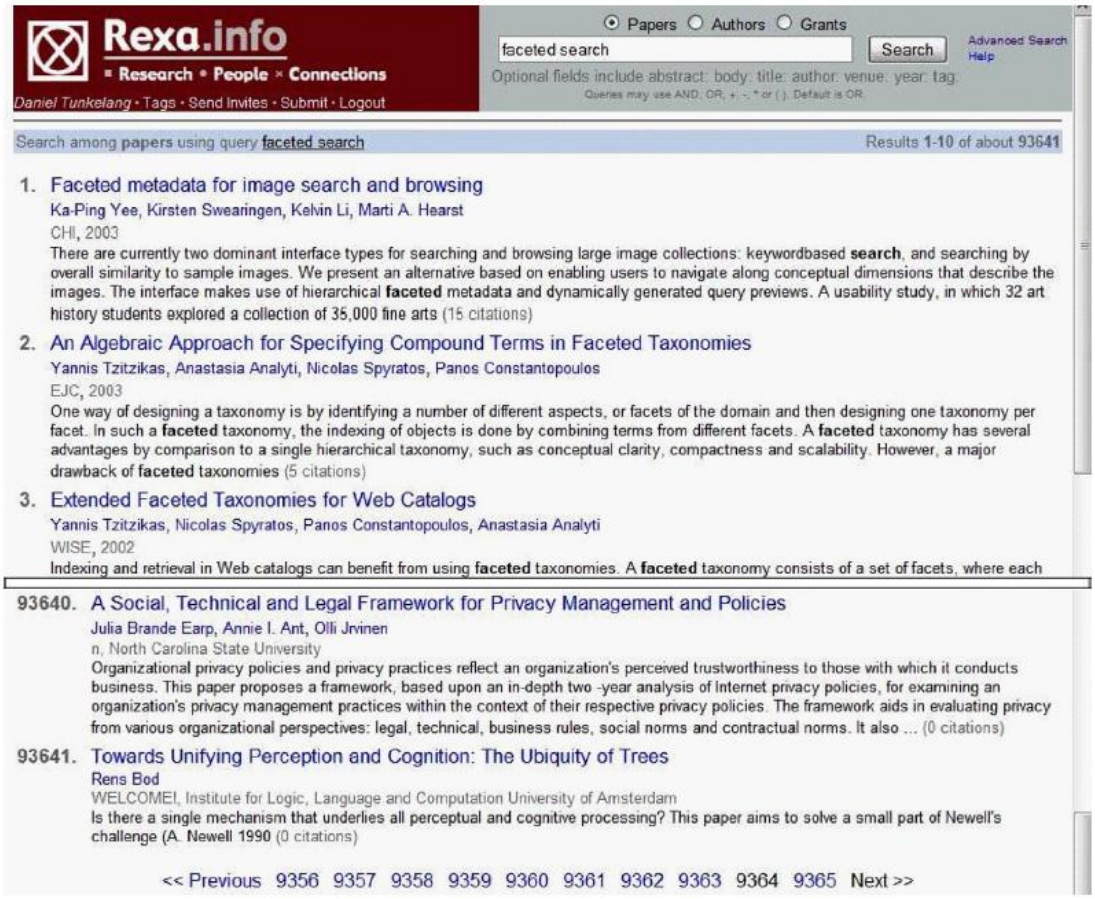
Εάν η ανάκτηση πληροφοριών είναι πολύ δύσκολη για τους ειδικούς που αναζητούν πληροφορίες, ποια είναι η εναλλακτική λύση; Η εναλλακτική λύση, η οποία είναι δημοφιλής στη χρήση από τις σύγχρονες μηχανές αναζήτησης, είναι η ανάκτησης βάση κατάταξης, γνωστή και ως Ranked Retrieval. Το 1961, ο Calvin Mooers [30], στου οποίου τη φήμη περιλαμβάνεται η συμβολή του στην καθιέρωση του όρου ανάκτηση πληροφοριών, εξέφρασε τη δυσαρέσκειά του στο μοντέλο ανάκτησης του Boolean και συνέβαλε στη δημιουργία ενός διαφορετικού συστήματος το οποί βασίζεται στο γνωμικό ότι [31]:

Είναι μια συνηθισμένη πλάνη, η οποία υπάρχει αυτή τη στιγμή και στην οποία γίνεται επένδυση πολλών εκατομμυρίων δολαρίων, ότι η άλγεβρα του George Boole είναι ο κατάλληλος φορμαλισμός για το σχεδιασμό των συστημάτων ανάκτησης. Η άποψη αυτή είναι τόσο ευρέως αποδεκτή όσο και λανθασμένη.

Κατά την αναζήτηση μιας εναλλακτικής λύσης στο μοντέλο ανάκτησης Boolean, οι ερευνητές ανάκτησης πληροφοριών υιοθέτησαν μια εντελώς διαφορετική προσέγγιση. Αντί να απαιτούν επίσημες δομημένες ερωτήσεις, ακολουθούσαν μια προσέγγιση βασισμένη σε μη δομημένα ερωτήματα ελεύθερου κειμένου, απελευθερώνοντας έτσι τους χρήστες από την ανάγκη να κατασκευάσουν σύνθετες εκφράσεις. Αντί να επιχειρούν να επιστρέψουν ένα ακριβές σύνολο αποτελεσμάτων στον χρήστη, χρησιμοποιούν ένα ευρύ πλαίσιο όρων και βασίζονται στην κατάταξη των αποτελεσμάτων ώστε να ευνοούνται τα πιο συναφή αποτελέσματα.

Η εικόνα 10 δείχνει ένα παράδειγμα ταξινόμησης της Rexa.info, μιας ψηφιακής βιβλιοθήκης και μιας μηχανής αναζήτησης που καλύπτει την επιστημονική βιβλιογραφία των επιστημών πληροφορικής. Ένα ερώτημα πολύ-επίπεδης αναζήτησης "faceted search" επιστρέφει ένα εντυπωσιακό αποτελέσματα της τάξης του 93.541. Ωστόσο, αυτά τα αποτελέσματα είναι για το ερώτημα 'Η' (OR) το οποίο έχει υψηλή ανάκληση, αλλά χαμηλή ακρίβεια. Οι προγραμματιστές εφαρμογών μετριάζουν αυτή τη χαμηλή ακρίβεια μέσω της κατάταξης με βάση την συνάφεια και πράγματι τα κορυφαία αποτελέσματα είναι πολύ πιο συναφή από αυτά που βρίσκονται στο κάτω μέρος της λίστας.

Τα ερωτήματα ελεύθερου κειμένου είναι, χωρίς αμφιβολία, πιο εύκολο να δημιουργούν, για τους χρήστες από τις τυπικές εκφράσεις Boolean. Αλλά αυτή η δυνατότητα έρχεται με ένα κόστος: το ερώτημα δεν αντιπροσωπεύει πλέον ένα καλά καθορισμένο φίλτρο ώστε να τεθεί ένα καλά καθορισμένο ερώτημα στη συλλογή εγγράφων. Αντίθετα, το ερώτημα γίνεται στόχος και η δυαδική έννοια ενός εγγράφου που ταιριάζει με το ερώτημα χαλαρώνει στο μέτρο της ομοιότητας. Ένα σύστημα ανάκτησης πληροφοριών βασισμένο σε μια τέτοια προσέγγιση δεν φιλτράρει πλέον έγγραφα αλλά περισσότερο τα ταξινομεί ανάλογα με το βαθμό στον οποίο ταιριάζουν με το ερώτημα.



Εικόνα 10 Αποτελέσματα για faceted search στο rexa.info

Το μοντέλο ανάκτησης μέσω κατάταξης οφείλει την επιτυχία του σε μεγάλο βαθμό σε δύο προσωπικότητες οι οποίοι και εργάσθηκαν πάνω στη σύγχρονη ανάκτηση πληροφοριών, τους: Gerald Salton και τον Karen Spärck Jones.

Η συμβολή του Gerald Salton στην ανάκτηση πληροφοριών είναι πολύ μεγάλη, λόγο αυτής έλαβε την υψηλότερη τιμή για την έρευνα στην ανάκτησης πληροφοριών για το έργο του. Ως εκ τούτου δίδεται βραβείο το οποίο φέρει και το όνομά του: το βραβείο Gerald Salton. Η ιδιαίτερη συμβολή του που σχετίζεται με το πεδίο έρευνας της παρούσας εργασίας, είναι το μοντέλο διανυσματικού διαστήματος (vector space model) [32]. Το vector space model αντιπροσωπεύει κάθε έγγραφο κειμένου ως διάνυσμα λέξεων, ή γενικότερα, όρων που μπορεί να περιλαμβάνουν φράσεις πολλαπλών λέξεων ή ρίζες λέξεων. Κάθε στοιχείο ενός διανυσματικού διαστήματος αντιπροσωπεύει, τουλάχιστον στη θεωρία, τη σχετική ισχύ του αντίστοιχου όρου, δηλαδή του βαθμού στον οποίο το έγγραφο αφορά τον όρο αυτό. Τώρα χρειαζόμαστε έναν τρόπο για να καθορίσουμε τις τιμές αυτών των στοιχείων η αλλιώς τα βάρη του όρου.

Το έργο αυτό ανέλαβε η Karen Spärck Jones, μια εξέχουσα προσωπικότητα στην ιστορία της ανάκτησης πληροφοριών. Πρότεινε μια στατιστική ερμηνεία της εξειδίκευσης των όρων (term specificity) που θα μπορούσε στη συνέχεια να εφαρμοστεί στο μοντέλο διανυσματικού χώρου [33]. Αυτό το σχήμα στάθμισης είναι γνωστό σήμερα ως συχνότητα όρων-αντίστροφη συχνότητα εγγράφου (term frequency-inverse document frequency tf-idf.) Το παύλα είναι ένα ατυχές γραμματικό συμβάν, καθώς το tf-idf στην πράξη πολλαπλασιάζει τους δύο παράγοντες και επομένως θα έπρεπε να γράφεται tf * idf.

Ο πρώτος παράγοντας, η συχνότητα όρων, είναι ο αριθμός των φορών που ένας συγκεκριμένος όρος εμφανίζεται σε συγκεκριμένο έγγραφο, διαιρούμενο με τον αριθμό των όρων που υπάρχουν στο έγγραφο, για να επιτευχθεί η κανονικοποιημένη βαθμολογία, η οποία παίρνει τιμές μεταξύ μηδέν και ενός. Οι όρου που έχουν μεγαλύτερη συχνότητα εμφάνισης υποδεικνύουν ότι οι όροι αυτοί είναι πιο αντιπροσωπευτικοί του περιεχομένου του εγγράφου, οι υπόλοιποι θεωρούνται ίσοι.

Δεν είναι δεδομένο, ότι οι υπόλοιποι όροι είναι ίσοι. Ο δεύτερος παράγοντας, η αντίστροφη συχνότητας εγγράφου, τονίζει σπάνιους όρους σε σχέση με τους συνηθισμένους. Οι όροι που εμφανίζονται σε λιγότερα έγγραφα σε μια συλλογή έχουν μεγαλύτερο βάρος από αυτούς που εμφανίζονται στα περισσότερα έγγραφα, επειδή η σπανιότητα συνεπάγεται κάτι το συγκεκριμένο. Η συχνότητα εγγράφου (document frequency) ενός όρου είναι το κλάσμα των εγγράφων της συλλογής που το περιέχει. Στην πραγματικότητα, παρά το όνομα, το idf συνήθως υποδηλώνει τον λογάριθμο αυτής της αντιστροφής.

Δεδομένου του μοντέλου διανυσματικού χώρου και του tf-idf ως στατιστικών ερμηνειών της εξειδικευμένης όρων, μπορούμε να αντιμετωπίσουμε ένα ερώτημα αναζήτησης ως γεωμετρικό πρόβλημα προσδιορισμού της απόστασης κάθε εγγράφου σε μια συλλογή, από το ερώτημα αναζήτησης. Πιο τυπικά, αντιμετωπίζουμε τόσο το ερώτημα όσο και ένα έγγραφο σαν διανύσματα σε ένα ανώτερο γεωμετρικό χώρο και υπολογίζουμε το συνημίτονο της γωνίας μεταξύ των δύο διανυσμάτων. Μία μικρότερη γωνία συνεπάγεται μεγαλύτερη ομοιότητα και το συνημίτονο της γωνίας παρέχει μια βαθμολογία που μπορεί να χρησιμοποιηθεί για την κατάταξη.

Έχουν σημειωθεί πολλές εξελίξεις στην ταξινόμηση με βάση την κατάταξη από τις πρώτες προσπάθειες των Salton και Spärck Jones. Μια σημαντική πρόοδος ήταν η ανάπτυξη της λανθάνουσας σημασιολογικής ανάλυσης (latent semantic analysis), μιας τεχνικής που εφαρμόζει την αποδόμηση της μοναδικής τιμής στον πίνακα όρων-εγγράφων για να ανακαλύψει τα κρυμμένα ή λανθάνοντα θέματα που αποτελούν τη βάση για ένα περιορισμένο διαστάσεων χώρο από το αρχικό διανυσματικό χώρο.

Ωστόσο, η πιο σημαντική πρόσφατη εξέλιξη στα μοντέλα κατάταξης στην ανάκτηση πληροφοριών δεν είχε καμιά σχέση με τα ερωτήματα εξαρτώμενα από μέτρα (query-dependent

measures), που συνδέονται περισσότερο με την έρευνα ανάκτησης πληροφοριών. Αντίθετα, η εμφάνιση του Παγκόσμιου Ιστού ως συλλογής εγγράφων στην οποία στοχεύουν περισσότερο οι μηχανές αναζήτησης οδήγησε τους ερευνητές να επικεντρωθούν γενικότερα σε συλλογές υπερκειμένων (hypertext) και στον παγκόσμιο ιστό ειδικότερα[34].

Η κοινωνική κατασκευή του ιστού οδήγησε τους ερευνητές να αναπτύξουν προσεγγίσεις κατάταξης που τονίζουν πριν από την αναζήτηση το γνωστικό αντικείμενο με το οποίο ασχολείται το κάθε έγγραφο. Η συγκεκριμένη τεχνική είναι γνωστή ως document priors και αντικατοπτρίζει την υπολογιστική αξία του εγγράφου πριν το ερώτημα. Τα δύο πιο αξιοσημείωτα μέτρα εγκυρότητας (authority) για τις συλλογές υπερκειμένου είναι ο αλγόριθμος HITS του Jon Kleinberg [35] και ο αλγόριθμος PageRank του Larry Page και του Sergey Brin [36], οι οποίοι χρησίμευσαν ως το αρχικό θεμέλιο της μηχανής αναζήτησης Google. Αν και η αρχή είναι ξεχωριστή από την έννοια της συνάφειας εγγράφων που προκάλεσε το μεγαλύτερο μέρος της εργασίας στην ταξινόμηση ανάκτησης. Ταιριάζει καλά στο πλαίσιο της κατάταξης των ταξινομημένων αποτελεσμάτων με σκορ που βασίζεται στην χρησιμότητα.

Η επιτυχία της τεχνικής ανάκτησης με κατάταξη σε σχέση με την τεχνική προκαθορισμένης ανάκτησης έχει τις ενστάσεις της ως προς τα πλεονεκτήματα επί της πρώτης τεχνικής, μιας και τα πλεονεκτήματα της πρώτης τεχνικής δεν έρχονται χωρίς ένα αξιοσημείωτο κόστος. Στο μοντέλο ανάκτησης μέσω κατάταξης, δεν μπορούμε πλέον να αιτιολογούμε ποια αποτελέσματα ταιριάζουν ή δεν ταιριάζουν με το ερώτημα. Παρόλο που η απώλεια αυτή δεν φαίνεται να είναι σημαντική σε μια διεπαφή όπου οι χρήστες εξετάζουν μόνο τα κορυφαία αποτελέσματα αναζήτησης, θα διαπιστώσουμε ότι αυτή η έλλειψη μιας σαφούς διαχωριστικής γραμμής είναι ιδιαίτερα προβληματική για την πολύ-επίπεδη αναζήτηση. Το faceted search είναι, στην καρδιά του, μια μέθοδος ανάκτησης προσανατολισμένη προς το σύνολο.

3.7 Πολύ-επίπεδη αναζήτηση πληροφοριών (faceted search)

Συγκεντρώνοντας όσα έχουμε μελετήσει ως τώρα, είδαμε ότι η πολύ-επίπεδη ταξινόμηση αντιμετωπίζει ορισμένους από τους περιορισμούς μιας ταξινομίας για την εκπροσώπηση της γνώσης. Θα δούμε πώς μπορούμε να δημιουργήσουμε καλύτερες διεπαφές για την ανάκτηση πληροφοριών χρησιμοποιώντας την πολύ-επίπεδη ταξινόμηση.

Πριν μελετήσουμε την πολύ-επίπεδη αναζήτηση, θα εξετάσουμε τους προκατόχους της: την παραμετρική αναζήτηση και την πολύ-επίπεδη πλοήγηση(faceted navigation). Κανένας από αυτούς τους προκάτοχους δεν λαμβάνει υπ’ όψην την κειμενική πλευρά των εγγράφων.

3.7.1 Παραμετρική αναζήτηση

Σαν παράδειγμα, θα χρησιμοποιήσουμε έναν τομέα δημοφιλή στους ερευνητές ανάκτησης πληροφοριών: το κρασί. Οι όψεις (facet) που τυπικά χρησιμοποιούνται για τον χαρακτηρισμό του οίνου, περιλαμβάνουν χαρακτηριστικά όπως (τον τύπο σταφυλιού, π.χ. Merlot), vintage (το έτος κατά το οποίο παράγεται ένα κρασί), περιοχή, βαθμολογία, τιμή ... Πώς μπορούμε να εφαρμόσουμε μια πολύ-επίπεδη αναπαράσταση των δεδομένων για να βοηθήσουμε κάποιον χρήστη να βρει ένα κρασί που ικανοποιεί τα ιδιαίτερα του γούστα;

Μια παραμετρική διεπαφή αναζήτησης είναι ουσιαστικά μια διεπαφή αναζήτησης τύπου Boolean για μια πολύπλευρη συλλογή: επιτρέπει στους χρήστες να διατυπώνουν ερωτήματα ορίζοντας οπτικά ένα σύνολο περιορισμών στις δοθέντες τιμές. Ένα ερώτημα αποτελεί ένα AND όλων των ORs: οι τιμές που επιλέγονται μέσα σε μία μόνο όψη συνδυάζονται χρησιμοποιώντας λογική OR, ενώ οι περιορισμοί που σχετίζονται με διαφορετικές πτυχές συνδυάζονται χρησιμοποιώντας μια λογική AND. Το σύστημα ανταποκρίνεται σε ένα ερώτημα με το σύνολο αντικειμένων στη συλλογή που το ικανοποιεί.

Ας χρησιμοποιήσουμε ένα συγκεκριμένο παράδειγμα για να δούμε πώς λειτουργεί η παραμετρική αναζήτηση. Υποθέτουμε πως έχουμε ένα χρήστη που ενδιαφέρεται για κόκκινα κρασιά από τη Γαλλία με βαθμολογία τουλάχιστον 90 και τιμή κατ' ανώτατο όριο 10 $. Η παραμετρική αναζήτηση του επιτρέπει να καθορίσει αυτό το ερώτημα, εκφράζεται ως το ακόλουθο σύνολο περιορισμών: {Ποικιλίες: Κόκκινο, Περιοχή: Γαλλία, Αξιολόγηση: 90, Τιμή: 10 $}. Μια παραμετρική διεπαφή αναζήτησης εμφανίζει ένα facet για κάθε ανεξάρτητη τιμή.

Όπως θα πρέπει να καταστεί σαφές και από αυτό το απλό παράδειγμα, διαφορετικά είδη facets, παράγουν διαφορετικά σύνολα αποτελεσμάτων.

Για ένα facet που έχει ονομαστικές τιμές (δηλ. Μια λίστα απαριθμούμενων κατηγοριών τιμών), έχει νόημα ο χρήστης να βλέπει μια λίστα επιλογών και να επιλέγει μία ή περισσότερες από αυτές ξεχωριστά. Εάν ο κατάλογος είναι μεγάλος τότε είναι απαραίτητη περαιτέρω προσπάθεια για την αποφυγή της υπερφόρτωσης των πληροφοριών.

Για ένα ιεραρχικό facet, όπως στο παράδειγμά μας, ο χρήστης μπορεί να επιλέξει μια τιμή που δεν βρίσκεται σε επίπεδο φύλλου, όπως το κόκκινο, ή μια τιμή ποικιλίας, όπως το Merlot. Ο χρήστης μπορεί να βλέπει ταυτόχρονα όλες τις επιλογές ή να πλοηγείτε από την κορυφή προς τα κάτω μέσω της ιεραρχίας.

Για αριθμητικά facet, όπως η τιμή ή η βαθμολογία, ο χρήστης πιθανότατα θέλει να επιλέξει μια περιοχή, ενδεχομένως χωρίς περιορισμούς όψεων. Στην οπτικοποιημένη διεπαφή, δεν υπάρχει καμιά καθοδήγηση σχετικά με τα λογικά όρια. Μια πιο εξελιγμένη διεπαφή μπορεί να προσφέρει τέτοια καθοδήγηση. Παρόλα αυτά, τονίζουμε ορισμένα από τα επιπρόσθετα

ζητούμενα, ώστε να τονίσουμε τον τρόπο με τον οποίο ο σχεδιασμός της διεπαφής αποτελεί κρίσιμη πτυχή, σε οποιαδήποτε εφαρμογή που θέτει ερωτήματα σε μια πολύ-επιπεδη συλλογή.

Σημαντικό είναι ότι η παραμετρική αναζήτηση είναι μια μορφή ανάκτησης μέσω κανόνων: προσφέρει ένα υποσύνολο λειτουργιών αναζήτησης Boolean, αν και λειτουργεί με facet και όχι από αδόμητο κείμενο. Όπως και η Boolean αναζήτηση σε κείμενο, δεν έχει αντιμετωπίζει το πρόβλημα ότι οι χρήστες δυσκολεύονται να διατυπώσουν τα ερωτήματά τους. Ενέχουν ένα πρόβλημα "εκατομμυρίων ή κανενός": τα μη προσδιορισμένα ερωτήματα επιστρέφουν πάρα πολλά αποτελέσματα, ενώ τα υπερβολικά ορισμένα ερωτήματα δεν επιστρέφουν αποτελέσματα. Η παραμετρική αναζήτηση προσφέρει εκφραστικότητα, αλλά δεν προσφέρει καθοδήγηση στους χρήστες μέσω του χώρου των πιθανών ερωτημάτων.

3.7.2 Faceted navigation

Το faceted navigation συμπληρώνει το κομμάτι που λείπει από την παραμετρική αναζήτηση: την καθοδήγηση. Η παραμετρική αναζήτηση απαιτεί ο χρήστης να εκφράσει μια ανάγκη πληροφόρησης σαν ερώτημα, κάνοντας επιλογές σε όλες τις πτυχές του συγκεκριμένου αντικειμένου που αναζητεί. Αντίθετα, το faceted navigation επιτρέπει στον χρήστη να επεξεργάζεται προοδευτικά ένα ερώτημα, βλέποντας την επίδραση κάθε επιλογής σε μία όψη στις διαθέσιμες επιλογές, αλλά και σε άλλες πτυχές.

Για να αναδείξουμε τη διαφορά, ας επιστρέψουμε στο προηγούμενο παράδειγμα με αντικείμενο το κρασί. Ένας χρήστης μπορεί να θέλει να ξεκινήσει έχοντας έναν προϋπολογισμό $10. Αυτή η επιλογή περιορίζει τις άλλες πτυχές, για παράδειγμα, δεν υπάρχουν γαλλικά κρασιά με λιγότερα από $10. Περαιτέρω επιλογές στις πτυχές της ποικιλίας και της περιοχής, περιορίζουν τις επιλογές στις απροσδιόριστες όψεις, για παράδειγμα, επιλέγοντας την Ισπανία ως περιοχή εξαλείφει το Sauvignon Blanc ως επιλογή από τα facet . Τελικά, ο χρήστης επιλέγει να βλέπει όλα τα αποτελέσματα που ταιριάζουν με τους καθορισμένους περιορισμούς.

Το faceted navigation προσφέρει μια εμπειρία προοδευτικής επεξεργασίας ερωτήσεων. Από την πλευρά του χρήστη, το faceted navigation εξαλείφει τα "αδιέξοδα" που μπορεί να προκύψουν από την επιλογή μη ικανοποιητικών συνδυασμών περιορισμού μεταξύ των facets. Στην πραγματικότητα, οι περισσότεροι συνδυασμοί των τιμών δεν είναι ικανοποιητικοί, επειδή το σύνολο των αποδεκτών συνδυασμών είναι τυπικά ένα αραιό υποσύνολο του συνόλου όλων των πιθανών συνδυασμών. Επομένως, το faceted navigation αντιμετωπίζει το πρόβλημα της παραμετρικής αναζήτησης "εκατομμύρια ή κανένα".

Όμως, δεν προσφέρονται όλα τα είδη facet για διεπαφές faceted navigation. Για παράδειγμα, στη διεπαφή που αναφέρουμε παραπάνω, ο χρήστης θα μπορούσε ακόμη να καταλήξει σε αδιέξοδο επιλέγοντας κρασιά για λιγότερο από $ 1. Μια πιο εξελιγμένη διεπαφή χρήστη μπορεί να αποφύγει αυτή τη δυνατότητα, για παράδειγμα, διαιρώντας τις αριθμητικές τιμές σε

διακριτές περιοχές. Γενικά, το faceted navigation έχει νόημα μόνο για facet των οποίων οι αξίες μπορούν να παρουσιαστούν μέσω καταλόγων επιλογής.

Αλλά πώς χειριζόμαστε τα δεδομένα κειμένου; Αυτή η ερώτηση μας οδηγεί στην επόμενη ενότητα στην οποία θα μελετήσουμε το faceted search.

3.7.3 Faceted search

Προηγουμένως εξετάσαμε τις προσεγγίσεις ανάκτησης πληροφοριών που έχουν σχεδιαστεί για αναζήτηση σε συλλογές κειμένου. Αντίθετα, οι προσεγγίσεις παραμετρικής αναζήτησης και το faceted navigation που περιγράφονται δεν γνωρίζουν το αδόμητο κείμενο και αντιθέτως θεωρούν ότι τα έγγραφα είναι συλλογές τιμών σε ένα σύστημα ταξινόμησης με χρήση facet.

Στην πράξη, οι περισσότερες συλλογές εγγράφων που μας ενδιαφέρουν είναι ημιδομημένες. Ένα τυπικό έγγραφο περιέχει έναν συνδυασμό αδόμητου κειμένου και δομημένων χαρακτηριστικών. Τα δομημένα χαρακτηριστικά μερικές φορές ονομάζονται μεταδεδομένα. Πράγματι, τα μεταδεδομένα ενός εγγράφου αποτελούνται από δομημένα χαρακτηριστικά σχετικά με το έγγραφο, τα οποία, τουλάχιστον με λογική έννοια, αποθηκεύονται με το έγγραφο.

Όταν αυτό το δομημένο περιεχόμενο συμβαδίζει με ένα σύστημα ταξινόμησης με χρήση facet, μπορούμε να συνδυάσουμε την αναζήτηση κειμένου, που εφαρμόζεται στο αδόμητο περιεχόμενο κειμένου, με το faceted navigation του δομημένου περιεχομένου. Αυτή η προσέγγιση είναι η ουσία το faceted search.

Ας επιστρέψουμε στο παράδειγμα του κρασιού, με την διαφορά ότι αυτή τη φορά χρησιμοποιήσουμε ένα πλουσιότερο μοντέλο εγγράφου όπου κάθε κρασί συνδέεται επίσης με περιγραφικό κείμενο.

Στην νέα περίπτωση, ο χρήστης δεν ξεκινά χρησιμοποιώντας τα facet αλλά μάλλον εκτελεί αναζήτηση ελεύθερου κειμένου για τα κρασιά που περιέχουν τη λέξη "sophisticated" στην περιγραφή τους. Ο χρήστης χρησιμοποιεί στη συνέχεια το facet των τιμών για να περιορίσει αυτά τα αποτελέσματα σε κρασιά κάτω των $ 10. Το σύστημα επιστρέφει τους 15 οίνους που ταιριάζουν με αυτά τα φίλτρα, ταξινομημένα κατά αξιολόγηση (σε ένα άλλο facet). Ο χρήστης μπορεί να επεξεργαστεί περαιτέρω αυτό το ερώτημα επιλέγοντας τιμές από άλλα facet, όπως τον τύπο, τη χώρα κ.ο.κ.

Όπως και το faceted navigation, το faceted search εξαλείφει τις επιλογές μεταξύ των τιμών του επιπέδου που θα οδηγούσαν σε αδιέξοδα. Για παράδειγμα, το εύρος τιμών των 90-100 δεν είναι διαθέσιμο ως επιλογή και προφανώς πρέπει μειωθούν οι απαιτήσεις όταν ψάχνουμε για φθηνό, εκλεπτυσμένο κρασί.

Όπως αποδεικνύεται, το faceted search μοιάζει πολύ με το σκάκι - χρειάζονται μόνο λίγα λεπτά για να κατανοήσει κάποιος τους κανόνες αλλά και πολλά χρόνια για να μάθει να παίζει καλά το παιχνίδι.

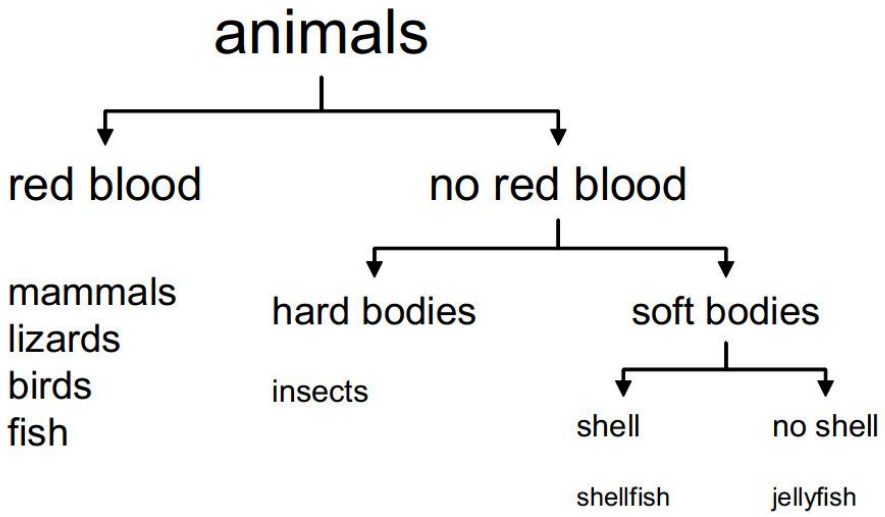
3.8 Ταξινομίες, οντολογίες

3.8.1 Ταξινομίες

Ο Αριστοτέλης δημιούργησε το πρώτο ολοκληρωμένο σύστημα φιλοσοφίας, που περιλαμβάνει την ηθική, την αισθητική, τη λογική, τις φυσικές επιστήμες, ακόμα και την πολιτική. Ο Αριστοτέλης ήταν από τους πρώτους που καθιέρωσαν ένα πλαίσιο γι 'αυτή τη συλλογική γνώση του ανθρώπινου είδους. Εάν όλη η φιλοσοφία είναι "μια σειρά υποσημειώσεων στον Πλάτωνα", τότε όλη η θεωρία και η πρακτική της εκπροσώπησης της γνώσης είναι σίγουρα μια σειρά υποσημειώσεων στον Αριστοτέλη.

Το σύστημα του Αριστοτέλη για την ταξινόμηση των ζωντανών πραγμάτων διαιρούσε τους οργανισμούς σε δύο ομάδες, φυτά και ζώα, διαιρώντας περαιτέρω τα ζώα σε αυτά "με αίμα" και "χωρίς αίμα". Τα είδη με αίμα, σε θηλαστικά και ωοτόκα και ούτω καθεξής εικόνα 11. Το Historia Animalium του Αριστοτέλη, το De Partibus Animalium και το De Generatione Animalium πλαισιώνουν την επιστήμη της ζωολογίας για τις επόμενες δύο χιλιετίες.

Ο Αριστοτέλης ήταν ο πρώτος ο οποίος όρισε τις ταξινομίες και ο ρόλος του αυτός θα ήταν ταυτόσημος με εκείνους που εργάζονται ως ταξινομητές σήμερα, οργανώνοντας τη γνώση σε ιεραρχίες. Η λέξη ταξινόμηση προέρχεται από την ελληνική (τάξη) που σημαίνει τάξη ή διάταξη και (νομός) που σημαίνει νόμος ή επιστήμη. Αναφερόταν αρχικά στην ταξινόμηση των ζωντανών οργανισμών, σύμφωνα με την ταξινομία του Αριστοτέλη που μόλις περιεγράφηκε τον 18ο αιώνα σαν Linnaean ταξινομία του φυσικού συστήματος (Systema Naturae) του 18ου αιώνα, που ονομάστηκε από τον συγγραφέα Carolus Linn.



Εικόνα 11 Υποσύνολο του συστήματος ταξινομίας ζώων του Αριστοτέλη

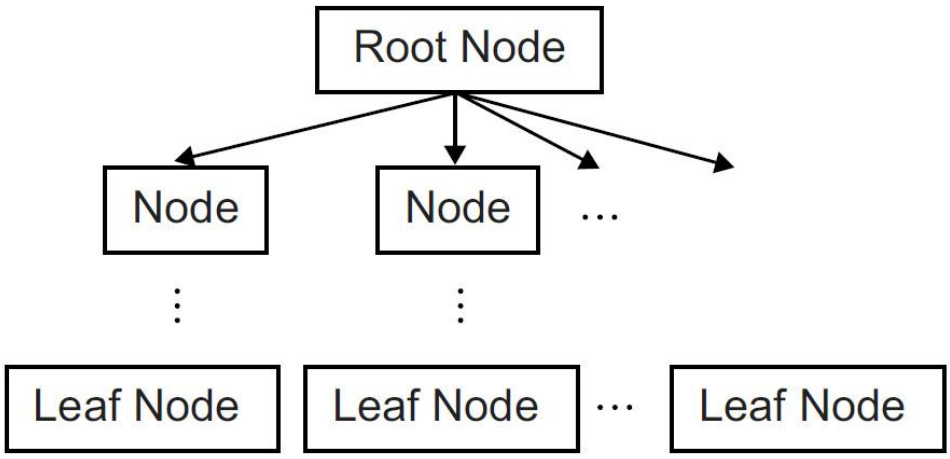
Σήμερα, ωστόσο, η λέξη ταξινομία αναφέρεται γενικότερα σε οποιοδήποτε σύστημα ιεραρχικής ταξινόμησης, καθώς και στις αρχές που διέπουν αυτά τα συστήματα (Εικόνα 15). Στη σύγχρονη χρήση, ταξινομία είναι κάθε οργάνωση πραγμάτων ή αφαίρεσης σε μια ιεραρχία ή μια δομή δέντρου. Χρησιμοποιώντας το δέντρο ως μεταφορά (αν και ένα δέντρο ανάποδα), υπάρχει ένας κόμβος ρίζας στην κορυφή, οι κόμβοι των φύλλων στο κάτω μέρος και κλάδους που συνδέουν κάθε γονικό κόμβο χωρίς φύλλα με τα παιδιά του (Εικόνα 15). Ένας γονέας μπορεί να έχει πολλά παιδιά, αλλά κάθε μη κόμβος έχει ακριβώς έναν γονέα.

Μια αναπαράσταση στην οποία ένας κόμβος μπορεί να έχει πολλαπλούς γονείς και έτσι πολλαπλές διαχωρισμένες διαδρομές που οδηγούν σε αυτόν από τη ρίζα ονομάζεται πολύ-ιεραρχικός. Η πολύ-ιεραρχία εισάγει επιπλέον δυνατότητες έκφρασης (π.χ. ένα μαχαιροπίρουνο μπορεί να είναι τόσο κουτάλι όσο και πιρούνι), εμφανίζοντας προφανώς μεγάλη πολυπλοκότητα. Επομένως, θα περιορίσουμε τις ταξινομίες τις οποίες θα μελετήσουμε σε όσες εμφανίζουν αυστηρές ιεραρχίες.

Ο ριζικός κόμβος σε μια ταξινομία αντιπροσωπεύει μια ταξινόμηση όλων των δεδομένων που περιγράφει ολόκληρη τη συλλογή αντικειμένων. Για παράδειγμα, στην ταξινομία του Αριστοτέλη, ο κόμβος ρίζας αντιστοιχεί στο σύνολο όλων των ζωντανών πραγμάτων. Τα παιδιά του αντιπροσωπεύουν τα ανώτατα τμήματα της συλλογής. Τα παιδιά τους, υποδιαιρέσεις αυτών και ούτω καθεξής, έως ότου οι κόμβοι των φύλλων αντιπροσωπεύουν τα ίδια τα αντικείμενα.

Η βασική ιδιότητα μιας ταξινομίας είναι ότι για κάθε αντικείμενο ή σύνολο αντικειμένων που αντιστοιχούν σε έναν κόμβο, υπάρχει ακριβώς μια μοναδική διαδρομή προς αυτόν από τον κόμβο ρίζας. Έτσι, μια ταξινομία επιβάλλει μια αυστηρή λογική διάταξη, για τη γνώση που αντιπροσωπεύει.

Τα σύγχρονα ανάλογα των ταξινομιών του Αριστοτέλη και του Linnaean περιλαμβάνουν τη δεκαεξαδική ταξινόμηση Dewey για τις βιβλιοθήκες, καθώς και τους καταλόγους ιστού όπως το Yahoo Directory και το Open Directory Project, και οι δύο από τους οποίους φιλοδοξούν να καταγράψουν μεγάλα υποσύνολα της τεράστιας συλλογής τοποθεσιών που είναι διαθέσιμες στον παγκόσμιο ιστό. Αξίζει να σημειωθεί ότι και στις τρεις περιπτώσεις, το αντικείμενο της ρύθμισης είναι να οργανωθούν αντικείμενα όπως π.χ. βιβλία στα ράφια, ιστοσελίδες σε μια λίστα και όχι να οργανωθούν αφηρημένες έννοιες.



Εικόνα 12Μια γενική ταξινομία

Η αυστηρή οργάνωση μιας ταξινομίας μπορεί να είναι πολύ άκαμπτη. Καθώς η παγκόσμια συλλογή γνώσεων έχει αυξηθεί, οι ταξινομίες προσπαθούν να διατηρήσουν αντίστοιχο ρυθμό. Συγκεκριμένα, η απαίτηση ότι κάθε κόμβος σε μια ταξινομία έχει μια μοναδική διαδρομή από τον ριζικό κόμβο, είναι ένας σκληρός περιορισμός, επιβάλλοντας στους σχεδιαστές ταξινομιών την υιοθέτηση αρκετών συμβιβασμών. Για παράδειγμα, η "Ευρωπαϊκή Ιστορία" είναι παιδί της "Ευρώπης" ή της "Ιστορίας"; Σε γενικές γραμμές, σύνθετες έννοιες αποτελούν πρόκληση για την ιεραρχική οργάνωση ταξινομιών.

Όπως αναφέραμε εν συντομία, οι πολύ-ιεραρχίες προσφέρουν λύση, αλλά η εισαγωγή και χρήση τους δημιουργεί περισσότερα προβλήματα από αυτά που λύνει, ειδικά για εκείνους που έχουν αναλάβει τη συντήρηση τους . Είναι αρκετά δύσκολο να διατηρηθεί μια τυποποιημένη ταξινομία, η δυσκολία γίνεται πολύ μεγαλύτερη στην περίπτωση που η μετακίνηση ενός κόμβου δεν μετακινεί απλώς παράλληλα και το αντίστοιχο υποδέντρο.[37]

3.8.2 Οντολογίες

Αρκετοί επιστήμονες της ανάκτησης πληροφοριών, που αντιμετωπίζουν εκφραστικά όρια του faceted search, έχουν στραφεί στις οντολογίες. Όπως και η ταξινομία, η λέξη οντολογία προέρχεται από το ελληνικό (οντό) νόημα του όντος και (logos) που σημαίνει επιστήμη, μελέτη, θεωρία. Στη φιλοσοφία, η οντολογία είναι η μελέτη της φύσης της ύπαρξης.

Ο Gruber [38] ορίζει "μια οντολογία ως ρητή προδιαγραφή μιας εννοιολογικής σκέψης. Οι οντολογίες συχνά εξομοιώνονται με τις ταξινομικές ιεραρχίες των τάξεων, τον ορισμό της τάξης και τη συσχέτιση υποκειμενικών σχέσεων, αλλά οι οντολογίες δεν χρειάζεται να περιορίζονται σε αυτές τις μορφές". που σημαίνει ότι μια οντολογία είναι μια εννοιολογική εκπροσώπηση της γνώσης και μπορεί να περιγράψει μια σειρά από έννοιες σε ένα συγκεκριμένο τομέα και τις σχέσεις μεταξύ τους. Στην πραγματικότητα, οι οντολογικές έννοιες δεν περιορίζονται σε λέξεις, αλλά επίσης θα μπορούσαν να είναι οντότητες, χαρακτηριστικά γνωρίσματα, κανόνες, περιορισμοί ή άλλοι τύποι πληροφοριών υψηλού επιπέδου.

Οι οντολογίες ταξινομούνται ως γενικές ή ειδικές για τον κάθε γνωστικό τομέα. Μια γενική οντολογία περιέχει ανεξάρτητες έννοιες από κάποιο γνωστικό τομέα και καλύπτει πολλαπλά θέματα, π.χ. YAGO2, OpenCyc [39], ώστε να έχει ευρύτερο φάσμα εφαρμογών. Οι ειδικές οντολογίες αντιπροσωπεύουν τη γνώση σε έναν συγκεκριμένο τομέα, π.χ. Οντολογία GENIA, Οντολογία γονιδίων, ανοιχτή βιολογική οντολογία, και λόγω της φύσης της ανεξαρτησίας, μια οντολογία συγκεκριμένου τομέα είναι δύσκολο να χρησιμοποιηθεί σε άλλους τομείς γνώσης. Λαμβάνοντας υπόψη τη λειτουργικότητα, εισήχθη μια τυπική ταξινόμηση οντολογιών από τον Guarino [40] ο οποίος διακρίνει τέσσερις τύπους οντολογίας: οντολογία ανώτατου τύπου, οντολογία γνωστικού τομέα, οντολογία εργασιών και οντολογία εφαρμογής. Μια οντολογία ανώτατου επιπέδου επιχειρεί να περιγράψει γνώση ανεξάρτητη του επιμέρους γνωστικού τομέα, η οντολογία εφαρμογής περιγράφει ένα συγκεκριμένο κλάδο ή μια εργασία.

Σε σύγκριση με κάποιους άλλους πόρους γνώσης, π.χ. εγκυκλοπαιδικές , ταξινομίες, κλπ μια οντολογία έχει υψηλότερη ικανότητα πληροφόρησης, μιας και παρέχει πληρέστερη κάλυψη γνώσης. Για παράδειγμα, η οντολογία YAGO2 περιέχει περισσότερα από 10 εκατομμύρια οντότητες και 120 εκατομμύρια γεγονότα. Το OpenCyc περιέχει γνώσεις από διάφορους πόρους, όπως το WordNet, το DBpedia και το Wikipedia [41].

Επιπλέον, οι οντολογίες έχουν σαφώς καθορισμένη δομή και παρέχουν εξεζητημένες γνώσεις. Για παράδειγμα, η λεξική οντολογία WordNet παρουσίασε δομές που δεν περιέχονται στα περισσότερα thesauruses και ταξινομίες, π.χ. ιεραρχίες hypernym, αντωνυμίες επίθετων και συνώνυμα ρημάτων. Επιπλέον, οι οντολογίες αναγνωρίζουν την εννοιολογική κατάσταση πίσω από τις λέξεις όπως π.χ. Το "London Eye" και το "Tower Bridge" δεν αναγνωρίζονται ως

συνώνυμα σε τυπικούς θησαυρούς, αλλά θα μπορούσαν να βρεθούν ως σχετικές έννοιες σε μια οντολογία γεωγραφίας.

Μία από τις προκλήσεις για πολλά συστήματα πληροφορικής που ασχολούνται με την εξατομίκευση σε διάφορους κλάδους εφαρμογών, όπως η ηλεκτρονική μάθηση, η υποστήριξη αποφάσεων, η ανάκτηση πληροφοριών, είναι η επιλογή μιας επίσημης μεθόδου για να μοντελοποιήσει αποτελεσματικά τη συμπεριφορά αλλά και τη γνώση του χρήστη. Πρόσφατες έρευνες δείχνουν ότι είναι επωφελής η χρήση βάσεων/οντολογιών γνώσης για την οργάνωση και τη διαχείριση της συμπεριφοράς και της γνώσης του χρήστη [42]. Η χρήση οντολογιών ανώτατου επιπέδου θα μπορούσε να είναι υπολογιστικά δαπανηρή καθώς οι περισσότερες οντολογίες ανώτατου επιπέδου έχουν πολύπλοκη δομή και περιέχουν γνώση σύνθετου τομέα. Αντίθετα, μια οντολογία εφαρμογών επικεντρώνεται σε έναν κλάδο ή σε μια συγκεκριμένη λειτουργία, που σημαίνει ότι είναι πιο κατάλληλη για ορισμένα πληροφοριακά συστήματα με πιο συγκεκριμένες απαιτήσεις.

Διάφορες ειδικές εφαρμογές συχνά απαιτούν τη δημιουργία μιας κατάλληλης οντολογίας. Γενικά, η δημιουργία οντολογιών είναι δύσκολη και δαπανηρή εργασία, καθώς είναι χρονοβόρα και κοστίζει σε εργατοώρες. Για παράδειγμα, η κατάρτιση μιας οντολογίας αναφορικά με το βιομηχανικό σχεδιασμό, πρέπει να βασίζεται σε γνώσεις σχεδίασης (αναγνωρισμένες από ειδικούς μηχανικούς) και σε γνωσιακές μελέτες, ενώ η οντολογία θα πρέπει να περιλαμβάνει έννοιες που σχετίζονται με τις λειτουργίες του προϊόντος, τις επιδόσεις, το υλικό, τις διαδικασίες παραγωγής, το περιβάλλον κ.λπ.[43]. Γενικά, η διαδικασία κατασκευής οντολογιών πρέπει να προσδιορίσει το πεδίο εφαρμογής του συστήματος και στη συνέχεια να ακολουθήσει τις απαιτήσεις του συστήματος και τις συστάσεις των εμπειρογνωμόνων του συγκεκριμένου τομέα. Μια άλλη μέθοδος κατασκευής είναι η ολοκλήρωση οντολογικών, η οποία ενσωματώνει μια σειρά από προκαθορισμένες οντολογίες μαζί για να ενισχύσει τη δυνατότητα γνώσης.

4

Google Natural Language Processing API

4.1 Εισαγωγή

Το API του Google NLP ενώνει άλλα προ-εκπαιδευμένα API μηχανικής μάθησης της Google όπως το Cloud Speech API, το οποίο είναι πλέον διαθέσιμο και σε δημόσια beta έκδοση, το Vision API και το Translate API.

Το API επεξεργασίας φυσικής γλώσσας βασιζόμενο στο Cloud υποστηρίζει επί του παρόντος κείμενα στα αγγλικά, ισπανικά και ιαπωνικά. Η Google με τον τρόπο αυτό προσφέρει μια υπηρεσία που μπορεί να καλύψει τις ανάγκες των προγραμματιστών και των επιχειρήσεων σε ένα ευρύ φάσμα βιομηχανιών για κλιμάκωσης και απόδοσης.[44][45][46]

Το Google NLP προσφέρει ένα API για ανάλυση συναισθήματος (sentiment) και αναγνώριση οντότητας (entity recognition). Το API συντακτικής ανάλυσης μπορεί να αναγνωρίσει τμήματα του λόγου POS και να δημιουργήσει δέντρα συσχέτισης εξαρτήσεων. Τα API αυτά έχουν την δυνατότητα να παρέχουν εξελιγμένες υπηρεσίες για ρομπότ συνομιλίας, βοηθώντας τα να κατανοήσουν τα εισερχόμενα ερωτήματα. Το NLP API αναλύει τη σημασία και τη δομή του κειμένου μέσω των εύχρηστων μοντέλων εκμάθησης μηχανών μέσω της διεπαφής REST API. Η λύση αυτή μπορεί να χρησιμοποιηθεί για την εξαγωγή σημαντικών πληροφοριών από έγγραφα κειμένων, άρθρα και ακαδημαϊκά κείμενα, ακόμα και να εντοπίσουμε το τμήμα το οποίο αφορά τα ‘συναισθήματα’ των πελατών σε εμπορικές εφαρμογές.

Σε αυτό το μέρος της εργασίας μας θα μελετήσουμε τα βασικά στοιχεία χρήσης του NLP API του Google Cloud. Αυτός ο εννοιολογικός οδηγός καλύπτει τους τύπους συναισθημάτων που μπορούν να γίνουν στο API φυσικής γλώσσας, με ποιο τρόπο κατασκευάσουμε αυτά τα αιτήματα και πώς χειριζόμαστε τις απαντήσεις τους.

4.2 Χαρακτηριστικά του Google NLP

Το API φυσικής γλώσσας έχει διάφορες μεθόδους για την εκτέλεση ανάλυσης και σχολιασμού των κειμένων. Κάθε επίπεδο ανάλυσης παρέχει πολύτιμες πληροφορίες για την κατανόηση της γλώσσας. Αυτές οι μέθοδοι παρατίθενται παρακάτω:

- **Η ανάλυση συναισθημάτων** εξετάζει το δεδομένο κείμενο και προσδιορίζει την επικρατούσα συναισθηματική γνώμη μέσα στο κείμενο, και ειδικότερα για να καθορίσει τη στάση του συγγραφέα ως θετική, αρνητική ή ουδέτερη. Η ανάλυση συναισθήματος πραγματοποιείται μέσω της μεθόδου analyzeSentiment.
- **Η ανάλυση οντοτήτων** ελέγχει το δεδομένο κείμενο για γνωστές οντότητες (τα κατάλληλα ουσιαστικά όπως δημόσια πρόσωπα, διακριτά σημεία κλπ. κοινά ουσιαστικά όπως εστιατόρια, γήπεδα κλπ.) Και επιστρέφει πληροφορίες σχετικά με αυτές τις οντότητες. Η ανάλυση οντότητας πραγματοποιείται με τη μέθοδο analyzeEntities.
- **Η ανάλυση του συναισθήματος οντοτήτων** εξετάζει το δεδομένο κείμενο για γνωστές οντότητες (τα κατάλληλα ουσιαστικά και κοινά ουσιαστικά), επιστρέφει πληροφορίες σχετικά με αυτές τις οντότητες και προσδιορίζει την επικρατούσα συναισθηματική άποψη της οντότητας μέσα στο κείμενο, ειδικά για να καθορίσει τη στάση ενός συγγραφέα προς την οντότητα ως θετική, αρνητικό ή ουδέτερη. Η ανάλυση οντότητας πραγματοποιείται με τη μέθοδο analyzeEntitySentiment.
- **Η συντακτική ανάλυση** εξάγει τις γλωσσικές πληροφορίες, διασπώντας το δεδομένο κείμενο σε μια σειρά φράσεων και tokens (γενικά, όρια λέξεων), παρέχοντας μια περαιτέρω ανάλυση για αυτές τα tokens. Η Συντακτική Ανάλυση πραγματοποιείται με τη μέθοδο analyzeSyntax.
- **Η ταξινόμηση περιεχομένου (Content classification)** αναλύει το περιεχόμενο κειμένου και επιστρέφει μια κατηγορία περιεχομένου για το ίδιο περιεχόμενο. Η ταξινόμηση περιεχομένου εκτελείται χρησιμοποιώντας τη μέθοδο classifyText.

Κάθε κλήση API ανιχνεύει και επιστρέφει τη γλώσσα, εάν η γλώσσα δεν καθορίζεται από τον προγραμματιστή στο αρχικό αίτημα.

Επιπλέον, αν θέλουμε να εκτελέσουμε πολλές λειτουργίες φυσικής γλώσσας σε δεδομένο κείμενο χρησιμοποιώντας μόνο μία κλήση API, το αίτημα annotateText μπορεί επίσης να χρησιμοποιηθεί για την εκτέλεση ανάλυσης συναισθήματος και ανάλυσης οντοτήτων.

4.2.1 Βασικά αιτήματα Google Natural Language

Το API φυσικής γλώσσας είναι ένα API REST και αποτελείται από αιτήματα και απόκρισεις τύπου JSON. Ένα απλό αίτημα ανάλυσης οντοτήτων φυσικής γλώσσας υπό μορφής JSON εμφανίζεται παρακάτω:

```
{
  "document":{
    "type":"PLAIN_TEXT",
    "language": "EN",
    "content":"'Lawrence of Arabia' is a highly rated film biography about \
            British Lieutenant T. E. Lawrence. Peter O'Toole plays \
            Lawrence in the film."
  },
  "encodingType":"UTF8"
}
```

Αυτά τα πεδία εξηγούνται παρακάτω:

- Το **έγγραφο (Document)** περιέχει τα δεδομένα για αυτό το αίτημα, το οποίο αποτελείται από τα ακόλουθα υπο-πεδία:
    - τύπος (**type**) εγγράφου τύπου (HTML ή PLAIN_TEXT)
    - γλώσσα (**language**)- (προαιρετικά) τη γλώσσα του κειμένου μέσα στο αίτημα. Εάν δεν έχει καθοριστεί, η γλώσσα εντοπίζεται αυτόματα. Για πληροφορίες σχετικά με τις γλώσσες που υποστηρίζονται από το API φυσικής γλώσσας, οι χρήστες μπορούν να ενημερωθούν στο τμήμα το οποίο αναφέρεται ως υποστήριξη γλώσσας. Οι μη υποστηριζόμενες γλώσσες θα επιστρέψουν ένα σφάλμα στην απάντηση JSON.
    - Είτε περιεχόμενο (**content**) είτε **gcsContentUri** που περιέχουν το κείμενο για αξιολόγηση. Εάν περάσουμε περιεχόμενο, αυτό το κείμενο περιλαμβάνεται απευθείας στο αίτημα JSON. Αν περάσουμε το **gcsContentUri**, το πεδίο πρέπει να περιέχει ένα URI που δείχνει περιεχόμενο κειμένου μέσα στο Google Cloud Storage.
- **encoding Type (Είδος κωδικοποίησης)** - (απαιτείται) θα πρέπει να υπολογιστεί το είδος κωδικοποίησης στο οποίο θα επιστραφούν οι αποκρήσεις, το οποίο πρέπει να ταιριάζει με την κωδικοποίηση του κειμένου προς ανάλυση.

4.3 Ανάλυση συναισθήματος (Sentiment analysis)

Η ανάλυση του συναισθήματος προσπαθεί να προσδιορίσει τη γενική στάση (θετική ή αρνητική) που εκφράζεται μέσα στο κείμενο. Το συναίσθημα αντιπροσωπεύονται από αριθμητικές τιμές βαθμολογίας και μεγέθους.

Πεδία απόκρισης ανάλυσης αισθήσεων

Ένα δείγμα αναλύει την ανταπόκριση του αισθήματος στο κείμενο της ‘Gettysburg Address’ φαίνεται παρακάτω:


```
{
  "documentSentiment":{
    "score":0.2,
    "magnitude":3.6
  },
  "language":"en",
  "sentences":[
    {
      "text":{
        "content":"Four score and seven years ago our fathers brought forth
      on this continent a new nation, conceived in liberty and dedicated to
      the proposition that all men are created equal.",
        "beginOffset":0
      },
      "sentiment":{
        "magnitude":0.8,
        "score":0.8
      }
    },
    ...
 }
```


Αυτές οι τιμές πεδίου περιγράφονται παρακάτω:

- **Το documentSentiment** περιέχει το γενικό συναίσθημα του εγγράφου, το οποίο αποτελείται από τα ακόλουθα πεδία:
    - **Τη βαθμολογία του συναισθήματος** κυμαίνεται μεταξύ -1,0 (αρνητική) και 1,0 (θετική) και αντιστοιχεί στη συνολική συναισθηματική κλίση του κειμένου.
    - **Το μέγεθος** υποδεικνύει τη συνολική ισχύ του συναισθήματος (τόσο θετική όσο και αρνητική) μέσα στο δεδομένο κείμενο, μεταξύ 0,0 και + inf. Αντίθετα από το σκορ, το μέγεθος δεν είναι κανονικοποιημένο. Κάθε έκφραση συναισθημάτων μέσα στο κείμενο (τόσο θετική όσο και αρνητική) συμβάλλει στο μέγεθος του κειμένου (και έτσι τα μπλοκ κειμένου μπορούν να έχουν μεγαλύτερα μεγέθη).
- **Η γλώσσα** περιέχει τη γλώσσα του εγγράφου, αυτή είτε μεταβιβάστηκε με το αρχικό αίτημα, είτε ανιχνεύτηκε αυτόματα αν δεν υπήρχε.
    - **Οι προτάσεις** περιέχουν μια λίστα με τις προτάσεις που εξήχθησαν από το πρωτότυπο έγγραφο, το οποίο τις περιέχει:
    - **Το συναίσθημα** το οποίο περιέχει τις τιμές συναρτήσεων στάθμης προτάσεων που επισυνάπτονται σε κάθε πρόταση, οι οποίες περιέχουν τιμές βαθμολογίας και μεγέθους όπως περιγράψαμε παραπάνω.

Μια βαθμολογίας της Gettysburg Address της τάξης 0,2 δείχνει ένα έγγραφο που είναι ελαφρώς θετικό στο συναίσθημα, ενώ η τιμή 3,6 δείχνει ένα σχετικά συναισθηματικό έγγραφο, δεδομένου του μικρού του μεγέθους (περίπου μιας παραγράφου). Σημειώνουμε ότι η πρώτη πρόταση της Gettysburg Address περιέχει ένα πολύ θετικό αποτέλεσμα 0,8.

4.3.1 Ερμηνεία της αξίας της ανάλυσης συναισθήματος

Η βαθμολογία του ‘συναισθήματος’ ενός εγγράφου υποδηλώνει τη συνολική εικόνα από άποψη ‘συναισθημάτων’ ενός εγγράφου. Το μέγεθος του αισθήματος ενός εγγράφου υποδεικνύει πόσο συναισθηματικό περιεχόμενο υπάρχει μέσα στο έγγραφο και αυτή η τιμή είναι συχνά ανάλογη με το μήκος του εγγράφου.

Είναι σημαντικό να σημειωθεί ότι το Google NLP API υποδεικνύει διαφορές μεταξύ θετικών και αρνητικών συναισθημάτων σε ένα έγγραφο, αλλά δεν εντοπίζει συγκεκριμένα θετικά και αρνητικά συναισθήματα. Για παράδειγμα, " angry " και "sad" θεωρούνται και τα δύο αρνητικά συναισθήματα. Ωστόσο, όταν το φυσικό API αναλύει κείμενο που θεωρείται «angry» ή κείμενο που θεωρείται «sad», η απόκριση δείχνει μόνο ότι το συναίσθημα στο κείμενο είναι αρνητικό, όχι «angry» ή «sad».

Ένα έγγραφο με ουδέτερη βαθμολογία (περίπου 0,0) μπορεί να υποδηλώνει ένα έγγραφο χαμηλού ‘συναισθήματος’ ή μπορεί να υποδηλώνει μικτά συναισθήματα, με τόσο υψηλές θετικές όσο και αρνητικές τιμές που ακυρώνουν η μία την άλλη. Γενικά, μπορούν να χρησιμοποιηθούν τιμές μεγέθους για να αποσαφηνιστούν αυτές οι περιπτώσεις, καθώς τα πραγματικά ουδέτερα έγγραφα θα έχουν χαμηλή τιμή μεγέθους, ενώ μικτά έγγραφα θα έχουν υψηλότερές τιμές μεγέθους.

Κατά τη σύγκριση εγγράφων μεταξύ τους (ειδικότερα σε έγγραφα διαφορετικού μήκους), θα πρέπει να βεβαιωθούμε ότι χρησιμοποιούμε τις τιμές μεγέθους για να βαθμονομήσουμε τα αποτελέσματά μας, καθώς μπορούν να μας βοηθήσουν να μετρήσουμε το σχετικό ποσό συναισθηματικού περιεχομένου.

Ο παρακάτω πίνακας δείχνει ορισμένες τιμές δειγμάτων και τον τρόπο ερμηνείας τους:

Sentiment	Sample Values
Clearly Positive*	"score": 0.8, "magnitude": 3.0
Clearly Negative*	"score": -0.6, "magnitude": 4.0
Neutral	"score": 0.1, "magnitude": 0.0
Mixed	"score": 0.0, "magnitude": 4.0

Το "Clearly positive" και το "clearly negative" συναίσθημα ποικίλλει ανάλογα με τις περιπτώσεις χρήσης. Μπορεί να βρούμε διαφορετικά αποτελέσματα για το συγκεκριμένο σενάριο μας. Συνίσταται να ορισθεί ένα κατώτατο όριο σαν βάση και στη συνέχεια, να προσαρμόσουμε το όριο μετά από έλεγχο και επαλήθευση των αποτελεσμάτων. Για παράδειγμα, μπορούμε να ορίσουμε ένα όριο οποιασδήποτε βαθμολογίας πάνω από 0,25 ως σαφώς θετικό και στη συνέχεια να τροποποιήσουμε το κατώτατο όριο βαθμολογίας σε 0,15 μετά την ανασκόπηση των δεδομένων και των αποτελεσμάτων μας και διαπιστώνοντας ότι οι βαθμολογίες από 0,15-0,25 θα πρέπει να θεωρηθούν θετικές.

4.4 Ανάλυση οντοτήτων

Η ανάλυση οντοτήτων παρέχει πληροφορίες σχετικά με οντότητες του κείμενου, οι οποίες γενικά αναφέρονται σε ονοματισμένα (named) πράγματα όπως διάσημα πρόσωπα, διακριτά αντικείμενα, κοινά αντικείμενα.

Οι οντότητες κατηγοριοποιούνται ευρέως σε δύο κατηγορίες:

1. Τα κατάλληλα ουσιαστικά που υποδηλώνουν μοναδικές οντότητες (συγκεκριμένους ανθρώπους, τόπους κ.λπ.)
2. Τα κοινά ουσιαστικά ονόματα (επίσης ονοματισμένα "ονομαστικά" στη φυσική γλώσσα). Μια καλή γενική πρακτική που ακολουθείται είναι ότι αν κάτι είναι ουσιαστικό, χαρακτηρίζεται ως "οντότητα". Οι οντότητες επιστρέφονται ως δεικτοποιημένες αντιστοιχήσεις στο αρχικό κείμενο.

Μια αίτηση ανάλυσης οντοτήτων θα πρέπει να περάσει ένα πεδίο αιτήματος encodingType, έτσι ώστε οι επιστρεφόμενες αντιστοιχήσεις να μπορούν να ερμηνευτούν σωστά.

4.4.1 Πεδία απόκρισης ανάλυσης οντοτήτων

Η ανάλυση οντοτήτων επιστρέφει ένα σύνολο ανιχνευμένων οντοτήτων και παραμέτρων που σχετίζονται με αυτές τις οντότητες, όπως ο τύπος της οντότητας, η συνάφεια της οντότητας με το συνολικό κείμενο και οι θέσεις στο κείμενο που αναφέρονται στην ίδια οντότητα. Οι οντότητες επιστρέφονται με τη σειρά των βαθμολογιών τους, που αντανακλούν τη συνάφεια τους (από την υψηλότερη στη χαμηλότερη) με το συνολικό κείμενο.

Μια απόκριση analyzeEntities σε αίτημα ανάλυσης οντοτήτων φαίνεται παρακάτω:


```
{  "entities": [
   {
    "name": "Lawrence of Arabia",
    "type": "WORK_OF_ART",
    "metadata": {
     "mid": "/m/0bx0l",
     "wikipedia_url": "http://en.wikipedia.org/wiki/Lawrence_of_Arabia_(film)"
    },
    "salience": 0.75222147,
    "mentions": [
     {
      "text": {
       "content": "Lawrence of Arabia",
       "beginOffset": 1
      },
    ] }],
 "language": "en" }
```

Να σημειωθεί ότι το API φυσικής γλώσσας επιστρέφει οντότητες για το "Lawrence of Arabia" (ταινία) και "T.E. Lawrence" (το πρόσωπο). Η ανάλυση οντοτήτων είναι χρήσιμη για την αποσαφήνιση παρόμοιων οντοτήτων όπως το "Lawrence" για την περίπτωση αυτή.

Τα πεδία που χρησιμοποιούνται για την αποθήκευση των παραμέτρων της οντότητας παρατίθενται παρακάτω:

- **type** υποδεικνύει τον τύπο αυτής της οντότητας (για παράδειγμα εάν η οντότητα είναι άτομο, τοποθεσία, καταναλωτικό αγαθό κ.λπ.). Αυτές οι πληροφορίες συμβάλλουν στη διάκριση ή / και αποσαφήνιση οντοτήτων και μπορούν να χρησιμοποιηθούν για τη σύνταξη σχεδίων ή την εξαγωγή πληροφοριών. Για παράδειγμα, η τιμή τύπου μπορεί να βοηθήσει στη διάκριση παρόμοιων οντοτήτων, όπως "Lawrence of Arabia", με ετικέτα WORK_OF_ART (ταινία), από το "T.E. Lawrence ", που έχει επισημανθεί ως PERSON.
- Τα **μεταδεδομένα** περιέχουν πληροφορίες πηγής σχετικά με το χώρο αποθήκευσης γνώσης της οντότητας. Αυτό το πεδίο μπορεί να περιέχει τα ακόλουθα υποπεδία:
    - Το **wikipedia_url**, αν υπάρχει, περιέχει τη διεύθυνση URL της Wikipedia που σχετίζεται με αυτήν την οντότητα.
    - Ο **MID (machine-generated identifier)**, εάν υπάρχει, περιέχει ένα αναγνωριστικό μηχανής (MID) που αντιστοιχεί στην καταχώριση του Google Knowledge Graph της οντότητας. Σημειώνεται ότι οι τιμές mid παραμένουν μοναδικές σε διάφορες γλώσσες, έτσι μπορούν να χρησιμοποιηθούν αυτές οι τιμές για να συνδεθούν οντότητες μαζί μεταξύ τους προερχόμενες από διαφορετικές γλώσσες.
- Η **σημασιολογία (salience)** υποδεικνύει τη σημασία ή τη συνάφεια αυτής της οντότητας με ολόκληρο το κείμενο του εγγράφου. Αυτός ο βαθμός μπορεί να βοηθήσει στην ανάκτηση πληροφοριών και τη σύνοψη, δίνοντας προτεραιότητα σε σημαντικές οντότητες. Τα αποτελέσματα πλησιέστερα στο 0.0 είναι λιγότερο σημαντικά, ενώ τα αποτελέσματα πιο κοντά στο 1.0 είναι πολύ σημαντικά.
- Οι **αναφορές** υποδεικνύουν θέσεις αντιστοιχίας εντός του κειμένου όπου αναφέρεται μια οντότητα. Αυτές οι πληροφορίες μπορεί να είναι χρήσιμες αν θέλουμε να βρούμε όλες τις αναφορές του προσώπου "Lawrence" στο κείμενο, αλλά όχι τον τίτλο της ταινίας. Μπορούμε επίσης να χρησιμοποιήσουμε αναφορές για τη συλλογή της λίστας των ψευδωνύμων, όπως "Lawrence", που αναφέρονται στην ίδια οντότητα "T.E. Lawrence ". Μια αναφορά οντότητας μπορεί να είναι ένας από τους δύο τύπους: PROPER ή COMMON. Μια ορθή οντότητα για τον "Lawrence της Αραβίας", για

παράδειγμα, θα μπορούσε να αναφερθεί άμεσα ως τίτλος ταινίας, ή ως ένα κοινό ουσιαστικό ("film biography" of T.E. Lawrence).

4.4.2 Ανάλυση συναισθημάτων οντοτήτων (Entity sentiment analysis)

Η ανάλυση του συναισθήματος των οντοτήτων συνδυάζει τόσο την ανάλυση οντοτήτων όσο και την ανάλυση συναισθημάτων και επιχειρεί να προσδιορίσει το συναίσθημα (θετικό ή αρνητικό) που εκφράζεται γύρω από τις οντότητες μέσα στο κείμενο. Το συναίσθημα της οντότητας αντιπροσωπεύεται από αριθμητικές τιμές βαθμολογίας και μεγέθους και καθορίζεται για κάθε αναφορά μιας οντότητας. Αυτές οι βαθμολογίες συγκεντρώνονται έπειτα σε μια συνολική βαθμολογία ‘αισθήματος’ και μέγεθος για μια οντότητα.

4.4.2.1 Αιτήματα ανάλυσης συναισθημάτων οντοτήτων

Τα αιτήματα Ανάλυσης Συναισθημάτων Οντότητας αποστέλλονται στο Google NLP API χρησιμοποιώντας τη μέθοδο analyzeEntitySentiment με την ακόλουθη μορφή:

```
{
  "document":{
    "type":"PLAIN_TEXT",
    "content":"I love R&B music. Marvin Gaye is the best.
          'What's Going On' is one of my favorite songs.
          It was so sad when Marvin Gaye died."
  },
  "encodingType":"UTF8"
}
```

Μπορεί να ορισθεί μια προαιρετική παράμετρος γλώσσας με το αίτημά μας, που να προσδιορίζει τον κώδικα γλώσσας για το κείμενο στην παράμετρο περιεχομένου. Εάν δεν καθορισθεί μια παράμετρος γλώσσας, τότε το API φυσικής γλώσσας ανιχνεύει αυτόματα τη γλώσσα για το περιεχόμενο της αίτησής σας.

4.4.2.2 Αποκρίσεις ανάλυσης συναισθημάτων οντοτήτων

Το Google NLP API επεξεργάζεται το δεδομένο κείμενο για να εξαγάγει τις οντότητες και να καθορίσει το συναίσθημα. Ένα αίτημα Ανάλυσης Συναισθημάτων Οντότητας επιστρέφει μια απάντηση που περιέχει τις οντότητες που βρέθηκαν στο περιεχόμενο του εγγράφου, μια αναφορά για κάθε φορά που αναφέρεται η οντότητα και οι αριθμητικές τιμές σκορ και μεγέθους

για κάθε αναφορά. Οι συνολικές τιμές βαθμολογίας και μεγέθους για μια οντότητα είναι το σύνολο των συγκεκριμένων τιμών βαθμολογίας και μεγέθους για κάθε αναφορά της οντότητας.


```
{
  "entities": [
    {
      "name": "R&B music",
      "type": "WORK_OF_ART",
      "metadata": {},
      "salience": 0.5306305,
      "mentions": [
        {
          "text": {
            "content": "R&B music",
            "beginOffset": 7
          },
          "type": "COMMON",
          "sentiment": {
            "magnitude": 0.9,
            "score": 0.9
          }
        }
      ],
      "sentiment": {
        "magnitude": 0.9,
        "score": 0.9
  ],
  "language": "en"
}
```

4.5 Συντακτική ανάλυση (Syntactic analysis)

Το Google NLP API παρέχει ένα ισχυρό σύνολο εργαλείων για την ανάλυση και την κατάτμηση του κειμένου μέσω της συντακτικής ανάλυσης. Για να εκτελεσθεί η συντακτική ανάλυση, χρησιμοποιείται η μέθοδος analyzeSyntax.

Η Συντακτική Ανάλυση αποτελείται από τις ακόλουθες λειτουργίες:

- **Εξαγωγή φράσης** (**Sentence extraction**) διασπάται η ροή του κειμένου σε μια σειρά φράσεων.
- **Tokenization** σπάει το ρεύμα του κειμένου σε μια σειρά από token, με το καθένα να αντιστοιχεί συνήθως σε μία μόνο λέξη.
- Το NLP API επεξεργάζεται τα tokens, χρησιμοποιώντας τις θέσεις τους μέσα στην προτάση, και προσθέτει συντακτικές πληροφορίες σε αυτά.

4.5.1 Αιτήματα συντακτικής ανάλυσης

Τα αιτήματα συντακτικής ανάλυσης αποστέλλονται στο API φυσικής γλώσσας μέσω της μεθόδου analyzeSyntax με την ακόλουθη μορφή:

```
{
  "document":{
    "type":"PLAIN_TEXT",
    "content":"Ask not what your country can do for you,
              ask what you can do for your country."
  },
  "encodingType":"UTF8"
}
```

4.5.2 Αποκρίσεις συντακτικής ανάλυσης

The Natural Language API processes the given text to extract sentences and tokens. A Syntactic Analysis request returns a response containing these sentences and tokens in the following form:

Το API φυσικής γλώσσας επεξεργάζεται το δεδομένο κείμενο για να εξάγει προτάσεις και tokens. Ένα αίτημα συντακτικής ανάλυσης επιστρέφει μια απόκριση που περιέχει αυτές τις προτάσεις και τα tokens με την ακόλουθη μορφή:

```
{
  "sentences": [
```

```
  ... Array of sentences with sentence information
 ],
 "tokens": [
  ... Array of tokens with token information
 ]
}
```

4.5.3 Εξαγωγή προτάσεων (Sentence extraction)

Κατά τη διεξαγωγή συντακτικής ανάλυσης, το API φυσικής γλώσσας επιστρέφει μια σειρά από προτάσεις που εξάγονται από το παρεχόμενο κείμενο, με κάθε πρόταση να περιέχει τα ακόλουθα πεδία μέσα σε ένα γονικό κείμενο:

- **beginOffset** που υποδεικνύει την μετατόπιση χαρακτήρων μέσα στο δεδομένο κείμενο όπου αρχίζει η πρόταση. Να σημειωθεί ότι αυτή η μετατόπιση υπολογίζεται με τη χρήση του δοσμένου τύπου κωδικοποίησης encodingType.
- Το περιεχόμενο (**content**) που περιέχει το πλήρες κείμενο της εξαγόμενης πρότασης.

Για παράδειγμα, λαμβάνεται το ακόλουθο στοιχείο προτάσεων για ένα αίτημα συντακτικής ανάλυσης του εγγράφου της Gettysburg Address:

```
{
 "sentences": [
  {
   "text": {
    "content": "Four score and seven years ago our fathers brought forth on
               this continent a new nation, conceived in liberty and
               dedicated to the proposition that all men are created
               equal.",
    "beginOffset": 0
   }
  },
  {
   "text": {
    "content": "Now we are engaged in a great civil war, testing whether
               that nation or any nation so conceived and so dedicated can
               long endure.",
    "beginOffset": 175
```

```
    }},
...
   {
    "text": {
     "content": "It is rather for us to be here dedicated to the great task
                remaining before us--that from these honored dead we take
                increased devotion to that cause for which they gave the
                last full measure of devotion--that we here highly resolve
                that these dead shall not have died in vain, that this
                nation under God shall have a new birth of freedom, and that
                government of the people, by the people, for the people
                shall not perish from the earth.",
      "beginOffset": 1002
    }}
 ],
 "language": "en"
}
```


Ένα αίτημα συντακτικής ανάλυσης στο Google NLP API περιλαμβάνει επίσης ένα σύνολο tokens. Μπορούν να χρησιμοποιηθούν οι πληροφορίες που σχετίζονται με κάθε διακριτικό tokens για την εκτέλεση περαιτέρω ανάλυσης σχετικά με τις προτάσεις που επιστρέφονται.

4.5.4 Tokenization

Η μέθοδος analyzeSyntax μετατρέπει το κείμενο σε μια σειρά από tokens, τα οποία αντιστοιχούν στα διαφορετικά στοιχεία του κειμένου. Η διαδικασία με την οποία το API αναπτύσσει αυτό το σύνολο μαρκών είναι γνωστό ως tokenization.

Μόλις γίνει η εξαγωγή των μαρκών, το Google NLP API επεξεργάζεται τα tokens για να καθορίσει το part of speech και lemma. Επιπλέον, τα tokens αξιολογούνται και τοποθετούνται μέσα σε ένα δέντρο περιγραφής εξαρτήσεων, το οποίο μας επιτρέπει να προσδιορίσουμε τη συντακτική έννοια των tokens και να δείξουμε τις μεταξύ τους σχέσεις και φράσεις. Οι συντακτικές και μορφολογικές πληροφορίες που σχετίζονται με τα tokens είναι χρήσιμες για την κατανόηση της συντακτικής δομής των προτάσεων.

The set of token fields returned in a syntactic analysis JSON response appears below:

Το σύνολο των πεδίων συμβολοσειράς που επιστρέφονται σε μια συντακτική ανάλυση τύπου JSON. Η απάντηση εμφανίζεται παρακάτω:

- Το κείμενο περιέχει τα δεδομένα κειμένου που σχετίζονται με αυτό το token, με τα ακόλουθα δευτερεύοντα πεδία:
    - Το **beginOffset** περιέχει την μετατόπιση (offset) των χαρακτήρων χαρακτήρων εντός του παρεχόμενου κειμένου.
    - Το **(Content)** περιέχει το πραγματικό περιεχόμενο κειμένου από το αρχικό κείμενο.
- Το **partOfSpeech** παρέχει γραμματικές πληροφορίες, συμπεριλαμβανομένων των μορφολογικών πληροφοριών, για το token, όπως ο χρόνος (γραμματική), το άτομο, ο αριθμός, το φύλο κλπ.
- Το λήμμα περιέχει τη λέξη ρίζα, η οποία μας επιτρέπει να κανονικοποιήσουμε τη χρήση λέξεων μέσα στο κείμενό μας. Για παράδειγμα, οι λέξεις " write ", " writing ", " wrote " και " written " βασίζονται στο ίδιο λήμμα ("write "). Επίσης, ο πληθυντικός και ενικός τύπος βασίζονται στο ίδιο λήμμα: οι λέξεις " house " και " houses " αναφέρονται και στην ίδια μορφή.
- Τα πεδία **dependenceEdge** προσδιορίζουν τη σχέση μεταξύ των λέξεων σε μια πρόταση που περιέχει tokens μέσω των άκρων σε κατευθυνόμενα δέντρο. Αυτές οι πληροφορίες μπορούν να είναι πολύτιμες για τη μετάφραση, την εξαγωγή πληροφοριών και την περίληψη. Κάθε πεδίο dependenceEdge περιέχει τα ακόλουθα δευτερεύοντα πεδία:
    - **Το headTokenIndex** παρέχει την τιμή του δείκτη του "γονικού token" του token που αναφέρεται στην πρόταση . Ένα token που δεν φέρει δείκτη αναφέρεται στον εαυτό του.
    - **Η ετικέτα Label** παρέχει τον τύπο της εξάρτησης του συγκεκριμένου token με το Token κεφαλής.

4.6 Κατηγοριοποίηση περιεχομένου (Content Classification)

Μπορούμε αν αναλύσουμε ένα έγγραφο μέσω του Google NLP και να λάβουμε ως απόκριση μια λίστα κατηγοριών περιεχομένου. Για να ταξινομήσουμε το περιεχόμενο σε ένα έγγραφο, καλούμε τη μέθοδο classifyText.

Το Google NLP API φιλτράρει τις κατηγορίες που επιστρέφονται με τη μέθοδο classifyText για να συμπεριλάβει μόνο τις σχετικές κατηγορίες για ένα αίτημα. Για παράδειγμα, αν οι κατηγορίες / Επιστήμη και / Επιστήμη / Αστρονομία ισχύουν και για ένα έγγραφο, τότε επιστρέφεται μόνο η κατηγορία / Επιστήμη / Αστρονομία, καθώς είναι το πιο συγκεκριμένο αποτέλεσμα.

4.7 Μέρος του λόγου (Parts of Speech)

Μέσα σε ένα συντακτικό αίτημα, οι πληροφορίες μορφολογίας και parts of speech (POS) επιστρέφονται στο πεδίο partOfSpeech της απόκρησης. Το πεδίο partOfSpeech περιέχει ένα σύνολο υπο-πεδίων με πληροφορίες για το POS καθώς και πιο ρητές μορφολογικές πληροφορίες. Αυτά τα υποπεδία αναφέρονται παρακάτω.

- **Η ετικέτα (Tag)** υποδηλώνει το γραμματικό μέρος της ομιλίας (Point of speech POS) (NOUN, VERB, κλπ.) και παρέχει πληροφορίες σύνταξης ανώτατου επιπέδου. Οι ετικέτες POS είναι χρήσιμες όταν θέλουμε να δημιουργήσουμε μοτίβα και / ή να μειώσουμε την ασάφεια (για παράδειγμα, "train" με ετικέτα NOUN έναντι VERB).
- Το **number** υποδηλώνει τον γραμματικό αριθμό μιας λέξης που υποδεικνύει τη διάκριση των αριθμών. Στα αγγλικά, η κατάληξη "s" χρησιμοποιείται συνήθως για να διακρίνει τους πληθυντικούς τύπους ουσιαστικών από τον ενικό. Για παράδειγμα, ορισμένες γλώσσες, όπως η αραβική, έχουν και την έννοια του διπλού αριθμού. Αυτό το πεδίο μπορεί να περιέχει τις ακόλουθες τιμές:
    - SINGULAR Ενικός.
    - PLURAL Πληθυντικός.
    - DUAL Και τα δυο (δυνατή επιλογή για ορισμένες γλώσσες).
- **Person** δηλώνει ένα γραμματικό πρόσωπο μιας λέξης, που δηλώνει τη σχέση ομιλητή με ένα γεγονός. Στην αγγλική γλώσσα, το άτομο χρησιμοποιείται συχνότερα σε αντωνυμίες για να γίνει διάκριση μεταξύ speakers (πρώτου προσώπου), those spoken to (δεύτερου προσώπου) και others (τρίτου προσώπου). Αυτό το πεδίο μπορεί να περιέχει τις ακόλουθες τιμές:
    - **FIRST** πρόσωπο δηλώνει το πρώτο πρόσωπο (ομιλητής)
    - **SECOND** άτομο δηλώνει το δεύτερο πρόσωπο (αυτός με τον οποίο ομιλεί το πρώτο).

      THIRD τρίτο πρόσωπο δηλώνει ένα "άλλο" πρόσωπο εκτός της συνομιλίας.

      REFLEXIVE_PERSON δηλώνει τη χρήση μιας αντανακλαστικής αντωνυμίας.
- Το **φύλο (gender)** υποδηλώνει ένα γραμματικό φύλο του ουσιαστικού. Αυτό το πεδίο μπορεί να περιέχει τις ακόλουθες τιμές:
    - FEMININE θηλυκό
    - MASCULINE αρσενικό
    - NEUTER ουδέτερο

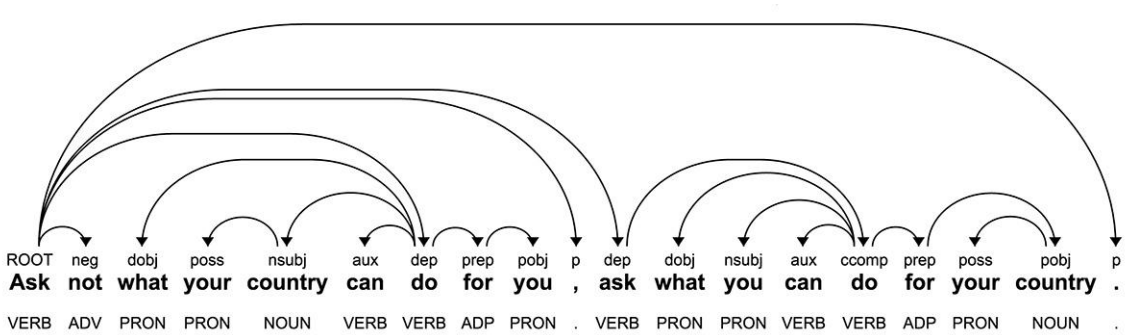
- **Γραμματικός χρόνος tense,** υποδηλώνει γραμματική ένταση ενός ρήματος, που υποδηλώνει την αναφορά του ρήματος σε μια χρονική στιγμή. Αυτό το πεδίο μπορεί να περιέχει τις ακόλουθες τιμές:
    - **CONDITIONAL_TENSE** is an alternate term for the more prevalent morphological term of "conditional mood." (See CONDITIONAL_MOOD below.)
    - **FUTURE** denotes an action taking place in the future. Note that in English, the future tense is most often denoted by adding the word "will" to a verb phrase.
    - **PAST** denotes an action taking place in the past.
    - **PRESENT** denotes an action taking place in the present.
    - **IMPERFECT** denotes an action taking place in the past, but which was not completed at that tense's frame of reference. Note that in English, the imperfect tense is most often denoted by adding a gerund form of a verb to the past tense as in "I was walking." An imperfect tense event takes place in the past, but is not completed relative to that past tense.
    - **PLUPERFECT** denotes an action that has taken place in the past, and was also completed at that tense's frame of reference. For example, "I had walked" takes place in the past, but was also complete during the past tense's frame of reference.
    - **CONDITIONAL_TENSE** υποθετικός όρος
    - **FUTURE** μέλλοντας
    - **PAST** παρελθοντικός
    - **PRESENT** ενεστώτας
    - **IMPERFECT** παρατατηκός
    - **PLUPERFECT** υπερσυντέλικος

4.8 Δέντρα εξαρτήσεων (Dependency trees)

Μέσα σε ένα αίτημα για συντακτική ανάλυση, οι πληροφορίες μορφολογίας και part-of-speech αναφέρονται στο πεδίο partOfSpeech της απόκρισης.

Για κάθε πρόταση μέσα στο κείμενο που αποστέλλεται για συντακτική ανάλυση, το API κατασκευάζει ένα δέντρο εξαρτήσεων που περιγράφει τη συντακτική δομή της πρότασης. Οι συντακτικές πληροφορίες επιστρέφονται στο πεδίο dependenceEdge της απάντησης.

Ένα διάγραμμα του δέντρου εξάρτησης για αυτή τη μοναδική πρόταση από την εναρκτήρια ομιλία του John F. Kennedy φαίνεται παρακάτω:



Εικόνα 13 Δέντρο εξαρτήσεων

5

Υλοποίηση

5.1 Εισαγωγή

Στο κεφάλαιο αυτό μελετάμε αρχικά το dataset, ακολούθως παρουσιάζουμε την μηχανή αναζήτησης Elasticsearch, τα χαρακτηριστικά και τα πλεονεκτήματα της, την χαρτογράφηση και τις παραμέτρους που χρησιμοποιήσαμε για την δημιουργία του index. Στην συνέχεια παραθέτονται οι σχετικές τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής. Εμβαθύνουμε ακολούθως στον τρόπο εξαγωγής γεγονότων, οντοτήτων και σχέσεων, στον ορθογραφικό έλεγχο, την ασαφή αναζήτηση, επέκταση ερωτημάτων, δημιουργία προτάσεων και στην εξαγωγή προτύπων αναζήτησης. Παρουσιάζεται η διεπαφή της εφαρμογής.

5.2 Σύνολο δεδομένων υλοποίησης (IMDB Dataset)

Στο παρών τμήμα της εργασίας μας, θα περιγράφει ο σχεδιασμός και η υλοποίηση ενός εργαλείου για την εξαγωγή των αρχείων δεδομένων IMDb και την εισαγωγή τους σε ένα index. Αυτή η προσέγγιση διαφέρει από άλλα δημοσιευμένα εργαλεία ή έρευνες στο ότι τα προηγούμενα έργα χρησιμοποιούν σχεσιακές βάσεις δεδομένων. Αυτή η προσέγγιση χρησιμοποιεί δομές δεδομένων προσανατολισμένες προς το έγγραφο.

Η Internet Movie Database (Διαδικτυακή βάση δεδομένων ταινιών), πιο γνωστή με την συντομογραφία IMDb, είναι διαδικτυακή βάση δεδομένων με πληροφορίες για ηθοποιούς, ταινίες, τηλεοπτικά προγράμματα, παρουσιαστές της τηλεόρασης, βιντεοπαιχνίδια και συντελεστές παραγωγής ταινιών ή προγραμμάτων. [47]

Η ανάπτυξη του έργου απαιτεί τη χρήση τεχνολογιών που τη δεδομένη στιγμή αποτελούν τεχνολογία αιχμής για προγραμματιστές ιστού και μηχανικούς λογισμικού. Τέλος, παρέχει μια διεπαφή ιστού για την εκτέλεση ερωτημάτων ενάντια στα δεδομένα εισαγωγής. Αυτά τα

ερωτήματα περιλαμβάνουν την αναζήτηση με ονόματα ανθρώπων, την αναζήτηση με ταινίες / τηλεοπτικούς τίτλους ή την προβολή συγκεκριμένων δεδομένων για ένα άτομο ή έναν τίτλο ταινίας / τηλεόρασης.

Το σύνολο δεδομένων περιέχει μεγάλο αριθμό ταινιών:

- 425.000+ τίτλους
- 1.700.000 + ηθοποιούς και συντελεστές
- Ταινίες από το 1891 μέχρι το παρόν
- Ξένες και ανεξάρτητες ταινίες, τηλεοπτικές ταινίες και εκπομπές, νέες κυκλοφορίες και πολλά άλλα.

Κάθε σύνολο δεδομένων περιλαμβάνεται σε αρχείο μορφοποιημένων με gzip, με διαχωρισμό με απόσταση (tab-separated-values TSV) κωδικοποιημένα στο σύνολο χαρακτήρων UTF-8. Η πρώτη γραμμή σε κάθε αρχείο περιέχει επικεφαλίδες που περιγράφουν τι υπάρχει σε κάθε στήλη. Το '\ N' χρησιμοποιείται για να δηλώσει ότι ένα συγκεκριμένο πεδίο λείπει ή είναι μηδενικό για τον εν λόγω τίτλο / όνομα. [48]

Τα αρχεία δεδομένων μπορούν να αποκτήσουν πρόσβαση και να μεταφορτωθούν από το https://datasets.imdbws.com/. Τα δεδομένα ανανεώνονται καθημερινά.

title.basics.tsv.gz

Περιέχει τις ακόλουθες πληροφορίες για τους τίτλους:

- **tconst (string)** - αλφαριθμητικό μοναδικό αναγνωριστικό του τίτλου
- **titleType (συμβολοσειρά)** - ο τύπος / μορφότυπος του τίτλου (π.χ. ταινία, σύντομη, τηλεοπτική σειρά, βίντεο, βίντεο κ.λπ.)
- **primaryTitle (συμβολοσειρά)** - ο πιο δημοφιλής τίτλος / ο τίτλος που χρησιμοποιείται από τους κινηματογραφιστές για διαφημιστικό υλικό στα σημεία προβολής
- **originalTitle (string)** - αρχικός τίτλος, στην αρχική γλώσσα
- **isAdult (boolean)** - 0: μη ενηλίκων τίτλος 1: τίτλος ενηλίκου
- **startYear (EEEE)** - αντιπροσωπεύει το έτος κυκλοφορίας ενός τίτλου. Στην περίπτωση τηλεοπτικών σειρών, είναι το έτος έναρξης σειράς
- **EndYear (YYYY)** - Σειρά τηλεοπτικών σειρών. '\ N' για όλους τους άλλους τύπους τίτλου
- **runtimeMinutes** - πρωταρχικός χρόνος εκτέλεσης του τίτλου, σε λεπτά
- **genres (σειρά συμβολοσειρών)** - Περιλαμβάνει έως και τρία είδη που σχετίζονται με τον τίτλο

name.basics.tsv.gz

Περιέχει τις ακόλουθες πληροφορίες για ονόματα:

- nconst (συμβολοσειρά) - αλφαριθμητικό μοναδικό αναγνωριστικό του ονόματος / προσώπου
- primName (συμβολοσειρά) - το όνομα με το οποίο το άτομο συνηθίζεται να αναφέρεται
- BirthYear - σε μορφή EEEE
- deathYear - σε μορφή EEEE, αν ισχύει, αλλιώς '\ N'
- primaryProfession (σειρά συστοιχιών) - τα κορυφαία 3 επαγγέλματα του ατόμου
- KnownForTitles (array of tconsts) - τίτλοι για τους οποίους είναι γνωστό το άτομο

5.3 Μηχανή αναζήτησης, Elasticsearch

Εισαγωγή

Το Elasticsearch είναι μια μηχανή αναζήτησης και ανάλυσης σε πραγματικό χρόνο. Επιτρέπει να εξερεύνηση των δεδομένων με ταχύτητα και σε κλίμακα που δεν ήταν ποτέ δυνατόν. Χρησιμοποιείται για αναζήτηση πλήρους κειμένου, δομημένη αναζήτηση, ανάλυση, αλλά και για τον συνδυασμό των τριών παραπάνω. Ο Elasticsearch είναι ένας διακομιστής αναζήτησης Apache Lucene. Αναπτύχθηκε από τον Shay Banon και δημοσιεύθηκε το 2010. Η υποστήριξη του αυτή τη στιγμή υλοποιείται από την Elasticsearch BV.[49]

- Η Wikipedia χρησιμοποιεί το Elasticsearch για να παρέχει αναζήτηση πλήρους κειμένου με επισημανθέντα αποσπάσματα αναζήτησης και την παραγωγή προτάσεων search-as-you-type και προτάσεων τύπου did-you-mean.
- Ο Guardian χρησιμοποιεί το Elasticsearch για να συνδυάσει αρχεία καταγραφής επισκεπτών με δεδομένα μέσων κοινωνικής δικτύωσης, για να παρέχει στους δημιουργούς του σε πραγματικό χρόνο την στοιχεία σχετικά με την ανταπόκριση του κοινού σε νέα άρθρα.
- Το Stack Overflow συνδυάζει την αναζήτηση πλήρους κειμένου με ερωτήματα γεωγραφικής κατανομής και χρησιμοποιεί παρόμοια άρθρα για να βρει σχετικές ερωτήσεις και απαντήσεις.
- Το GitHub χρησιμοποιεί το Elasticsearch για την αναζήτηση 130 δισεκατομμυρίων γραμμών κώδικα.

Το Elasticsearch είναι μια μηχανή αναζήτησης ανοιχτού κώδικα, η οποία είναι χτισμένη πάνω στο Apache Lucene [50][51], μια πλήρης βιβλιοθήκη μηχανών αναζήτησης. Η Lucene είναι

αναμφισβήτητα η πιο προηγμένη βιβλιοθήκη μηχανών αναζήτησης που λειτουργεί σήμερα και αποτελεί λογισμικό ανοικτού κώδικα, αλλά διαθέτει και διάφορες κλειστές παραλλαγές .

Η Lucene είναι απλώς μια βιβλιοθήκη, για να αξιοποιηθεί η δύναμή της, πρέπει να γίνει χρήση της Java και να ενσωματωθεί η Lucene απευθείας στην οποιαδήποτε εφαρμογή. Ακόμη θα χρειαστεί βαθύτερη γνώση της ανάκτηση πληροφοριών ώστε να γίνει κατανοητή η λειτουργία της. Το Elasticsearch γράφεται επίσης σε Java και χρησιμοποιεί το Lucene εσωτερικά για την ευρετηρίαση και τις αναζητήσεις, επίσης αποσκοπεί στην εύκολη αναζήτηση πλήρους κειμένου με την απόκρυψη της πολυπλοκότητας του Lucene πίσω από ένα απλό, συνεκτικό, RESTful (representational state transfer) API (application programming interface) καθώς το Lucene είναι αρκετά περίπλοκο.

Ωστόσο, το Elasticsearch είναι κάτι περισσότερο από το Lucene και πολύ περισσότερο από απλή αναζήτηση πλήρους κειμένου. Μπορεί να περιγράφει ως εξής:

- Ένα πραγματικού χρόνου κατανεμημένο σύστημα διαχείρισης εγγράφων, όπου κάθε πεδίο είναι ευρετηριασμένο και μπορεί να αναζητηθεί
- Μια κατανεμημένη μηχανή αναζήτησης με αναλυτικά στοιχεία σε πραγματικό χρόνο
- Παρέχει δυνατότητα κλιμάκωσης σε εκατοντάδες διακομιστές και επεξεργασίας τεράστιας ποσότητας δεδομένων τόσο δομημένων όσο και αδόμητων.
- Παρέχει όλες αυτές τις λειτουργίες σε ένα αυτόνομο διακομιστή που η κάθε εφαρμογή μπορεί να μιλήσει μέσω ενός απλού RESTful API, ανεξάρτητα σε ποια γλώσσα προγραμματισμού είναι γραμμένη η εφαρμογή.
- Μπορούν να γίνουν ερωτήματα ακόμη και από τη γραμμή εντολών.
- Είναι επεκτάσιμο και μπορεί να επεξεργαστεί και να διανύει τεράστια ποσότητα δεδομένων τόσο δομημένων όσο και αδόμητων.
- Το Elasticsearch μπορεί να χρησιμοποιηθεί ως αντικαταστάτης των δομών δεδομένων όπως το MongoDB και το RavenDB.
- Χρησιμοποιεί εξομαλύνσεις (denormalization) για να βελτιώσει την απόδοση αναζήτησης.
- Είναι ανοικτού κώδικα και διατίθεται υπό την έκδοση 2.0 της άδειας Apache.

Elasticsearch - Βασικές έννοιες

Οι βασικές έννοιες του Elasticsearch είναι οι εξής:

- **Κόμβος:** Αναφέρεται σε ένα ενεργό στιγμιότυπο (σέρβερ) του Elasticsearch. Αποτελεί μεμονωμένο φυσικός ή εικονικό διακομιστή που εξυπηρετεί πολλαπλούς κόμβους ανάλογα με τις δυνατότητες των φυσικών πόρων, όπως η μνήμη RAM, οι αποθηκευτικοί χώροι αλλά και η επεξεργαστική ισχύς.

- **Συστάδα (Cluster):** Είναι μια συλλογή από έναν ή περισσότερους κόμβους. Οι Clusters παρέχουν συλλογική ευρετηρίαση και δυνατότητες αναζήτησης σε όλους τους κόμβους για αναζήτηση σε ολόκληρη τη συλλογή.
- **Ευρετήριο(index):** Πρόκειται για μια συλλογή διαφορετικών τύπων εγγράφων και ιδιοτήτων εγγράφων. Ο index χρησιμοποιεί επίσης την έννοια των θραυσμάτων (Shards) για τη βελτίωση της απόδοσης. Για παράδειγμα, ένας index περιέχει δεδομένα μιας εφαρμογής κοινωνικής δικτύωσης.
- **Τύπος / χαρτογράφηση (Type/Mapping):** Μια συλλογή εγγράφων μοιράζονται ένα σύνολο κοινών πεδίων που υπάρχουν στον ίδιο ευρετήριο. Για παράδειγμα, ένας index περιέχει δεδομένα μιας εφαρμογής κοινωνικής δικτύωσης και στη συνέχεια μπορεί να υπάρχει ένας συγκεκριμένος τύπος δεδομένων προφίλ χρήστη, ένας άλλος τύπος για δεδομένα μηνυμάτων και ένα άλλο για σχόλια δεδομένα.
- Έγγραφο (document): Είναι μια συλλογή από πεδία με συγκεκριμένο τρόπο που ορίζεται σε μορφή JSON [52] στο **mapping**. Κάθε έγγραφο ανήκει σε έναν τύπο και εντοπίζεται μέσα σε έναν index. Κάθε έγγραφο συσχετίζεται με ένα μοναδικό αναγνωριστικό, το οποίο ονομάζεται UID.
- Shard(τμήμα): Οι index υποδιαιρούνται κάθετα σε Shard. Αυτό σημαίνει ότι κάθε Shard περιέχει όλες τις ιδιότητες του εγγράφου, αλλά περιέχει μικρότερο αριθμό αντικειμένων JSON από το ευρετήριο. Ο οριζόντιος διαχωρισμός κάνει το shard έναν ανεξάρτητο κόμβο, ο οποίος μπορεί να αποθηκευτεί σε οποιονδήποτε σημείο του cluster. Το primary shard είναι το αρχικό οριζόντιο shard ενός index, στη συνέχεια αυτά τα primary shard αναπαράγονται σε κομμάτια replicas.
- Replicas: Το Elasticsearch επιτρέπει στο χρήστη να δημιουργεί αντίγραφα των index και των shard. Η αναπαραγωγή όχι μόνο συμβάλλει στην αύξηση της διαθεσιμότητας δεδομένων σε περίπτωση βλάβης, αλλά και βελτιώνει την απόδοση της αναζήτησης πραγματοποιώντας παράλληλη αναζήτηση σε αυτά τα αντίγραφα.

5.3.1 Βασικές στοιχεία αρχιτεκτονικής του Elasticsearch

5.3.1.1 Nodes - Cluster

Ένας κόμβος είναι μια τρέχουσα παρουσία του Elasticsearch, ενώ ένα cluster αποτελείται από έναν ή περισσότερους κόμβους με το ίδιο cluster.name που δουλεύουν μαζί για να μοιραστούν τα δεδομένα και το φόρτο εργασίας τους. Καθώς οι κόμβοι προστίθενται ή αφαιρούνται από το cluster, αυτό αναδιοργανώνεται για να κατανείμει τα δεδομένα ομοιόμορφα. [53] [54]

Ένας κόμβος στο cluster εκλέγεται ως κύριος κόμβος, ο οποίος είναι υπεύθυνος για τη διαχείριση αλλαγών σε ολόκληρο το cluster, όπως τη δημιουργία ή τη διαγραφή ενός index ή την προσθήκη ή την αφαίρεση ενός κόμβου στο cluster. Ο κύριος κόμβος δεν χρειάζεται να συμμετέχει σε αλλαγές ή αναζητήσεις στο επίπεδο εγγράφων, πράγμα που σημαίνει ότι η κατοχή ενός μόνο κύριου κόμβου δεν θα δημιουργήσει συμφόρηση καθώς αυξάνεται η επισκεψιμότητα. Οποιοσδήποτε κόμβος μπορεί να γίνει και κύριος.

Σαν χρήστες, μπορούμε να μιλήσουμε με οποιονδήποτε κόμβο στο Elasticsearch, ακόμη και στον κύριο κόμβο. Κάθε κόμβος ξέρει πού αποθηκεύεται το κάθε έγγραφο και μπορεί να διαβιβάσει το αίτημά απευθείας στους κόμβους που κατέχουν τα δεδομένα που μας ενδιαφέρουν. Σε όποιο κόμβο και αν βρισκόμαστε αυτός μπορεί να διαχειρίζεται τη διαδικασία συγκέντρωσης των κειμένων της συλλογής από τον κόμβο ή τους κόμβους που κατέχουν τα δεδομένα. Ακόμη μπορεί να επιστρέφει την τελική απάντηση στον χρήστη. Όλη η διαχείριση εκτελείται με διαφάνεια από τον Elasticsearch.

Το σημαντικότερο στατιστικό στοιχείο που χρήζει παρακολούθησης είναι η καλή κατάσταση των ομάδων, η κατάσταση αυτή αναφέρεται είτε με πράσινο, είτε κίτρινο είτε κόκκινο.

- **Πράσινο**. Όταν όλα τα primary shards όπως και τα replicas είναι ενεργά.
- **Κίτρινο**. Όταν όλα τα primary shards είναι ενεργά, αλλά δεν είναι και όλα τα replicas shards ενεργά.
- **Κόκκινο**. Όταν δεν είναι όλα τα primary shards ενεργά.

5.3.1.2 Index - Shard - Replicas

Για να προσθέσουμε δεδομένα στην Elasticsearch, χρειαζόμαστε έναν index, ένα μέρος για την αποθήκευση των σχετικών δεδομένων. Στην πραγματικότητα, ένας index είναι απλώς ένας λογικός χώρος που δείχνει ένα ή περισσότερα μέρη του κειμένου-shards.

Ένα shard είναι μια μονάδα χαμηλού επιπέδου που κρατάει μόνο ένα τμήμα από όλα τα δεδομένα του index. Ένα shard είναι ένα στιγμιότυπο του Lucene και αποτελεί μια ολοκληρωμένη μηχανή αναζήτησης από μόνη του. Τα έγγραφά μας αποθηκεύονται και αναπροσαρμόζονται σε shard, αλλά οι εφαρμογές δεν επικοινωνούν μαζί τους άμεσα, αντ' αυτού, μιλάνε σε έναν index.

Τα shards στο Elasticsearch διανέμουν τα δεδομένα στο cluster μας, εάν τα φανταστούμε σαν τμήματα αποθηκευμένων δεδομένων. Τα έγγραφα αποθηκεύονται σε shards και τα shards κατανέμονται στους κόμβους του cluster. Καθώς το cluster μας αναπτύσσεται ή συρρικνώνεται, το Elasticsearch θα μετακινήσει αυτόματα τα shards μεταξύ των κόμβων έτσι ώστε το cluster να παραμείνει ισορροπημένο.

Ένα shard μπορεί να είναι είτε primary είτε replica. Κάθε έγγραφο του index ανήκει σε ένα μόνο primary shard, οπότε ο αριθμός των primary shards που έχουν καθορίσει είναι το μέγιστο ποσό δεδομένων που μπορεί να κρατήσει ο index μας.

Ένα replica shard είναι απλώς ένα αντίγραφο ενός primary τμήματος. Τα replica χρησιμοποιούνται για την παροχή περιττών αντιγράφων των δεδομένων μας για προστασία από αποτυχία υλικού και για την εξυπηρέτηση αιτημάτων ανάγνωσης όπως η αναζήτηση ή η ανάκτηση ενός εγγράφου.

Ο αριθμός των primary shards σε έναν index καθορίζεται τη στιγμή που, αλλά ο αριθμός των τμημάτων replica μπορεί να αλλάξει ανά πάσα στιγμή.

5.3.1.3 Έγγραφο - Document

Ένα αντικείμενο είναι μια δομή δεδομένων, συγκεκριμένης γλώσσας, εντός της μνήμης. Για να το στείλουμε μέσω του δικτύου ή να το αποθηκεύσουμε, πρέπει να μπορούμε να το απεικονίσουμε σε κάποια τυποποιημένη μορφή. Το JSON είναι ένας τρόπος αντιπροσώπευσης αντικειμένων σε κείμενο αναγνώσιμο από άνθρωπο. Έχει γίνει το καθιερωμένο πρότυπο για την ανταλλαγή δεδομένων στον κόσμο των συστημάτων NoSQL(non relational database). Όταν ένα αντικείμενο έχει κωδικοποιηθεί σε JSON (JavaScript Object Notation), είναι γνωστό ως έγγραφο JSON.

Το Elasticsearch αποτελεί ένα κατανεμημένο document store. Μπορεί σε πραγματικό χρόνο να αποθηκεύει και να ανακτά σύνθετες δομές δεδομένων κωδικοποιημένες ως έγγραφα JSON. Με άλλα λόγια, μόλις ένα έγγραφο έχει αποθηκευτεί στο Elasticsearch, μπορεί να ανακτηθεί από οποιονδήποτε κόμβο στο cluster.

Φυσικά, δεν χρειάζεται μόνο να αποθηκεύουμε δεδομένα πρέπει επίσης να εκτελούνται ερωτήματα μαζικά και με ταχύτητα. Παρόλο που υπάρχουν λύσεις NoSQL που μας επιτρέπουν να αποθηκεύουμε αντικείμενα ως έγγραφα, οι λύσεις αυτές απαιτούν ακόμα να προσδιορίσουμε το πώς θέλουμε να ανακτήσουμε τα δεδομένα μας και ποια πεδία απαιτούνται σε ένα ευρετήριο ώστε να γίνει γρήγορη ανάκτηση δεδομένων.

Στο Elasticsearch, όλα τα δεδομένα σε κάθε πεδίο είναι ευρετηριασμένα από προεπιλογή. Δηλαδή, κάθε πεδίο έχει έναν ειδικό ανεστραμμένο δείκτη για γρήγορη ανάκτηση. Σε αντίθεση με τις περισσότερες άλλες βάσεις δεδομένων, μπορεί να χρησιμοποιήσει όλους αυτούς τους ανεστραμμένους δείκτες στο ίδιο ερώτημα, για να επιστρέψει αποτελέσματα με εκπληκτική ταχύτητα.

Οι περισσότερες οντότητες ή αντικείμενα μπορούν να κωδικοποιηθούν σε ένα αντικείμενο JSON, που περιέχει κλειδιά και τιμές τους. Ένα κλειδί είναι το όνομα ενός πεδίου ή μιας ιδιότητας και μια τιμή μπορεί να είναι μια συμβολοσειρά, ένας αριθμός, ένα Boolean, ένα άλλο αντικείμενο, ένας πίνακας τιμών ή κάποιος άλλος εξειδικευμένος τύπος, όπως μια

συμβολοσειρά που αντιπροσωπεύει μια ημερομηνία ή ένα αντικείμενο που αντιπροσωπεύει γεωγραφική θέση πχ:

```
{
  "name":"John Smith",
  "age":42,
  "confirmed":true,
  "join_date":"2014-06-01",
  "home":{
    "lat":51.5,
    "lon":0.1
  },
  "accounts":[
    {
      "type":"facebook",
      "id":"johnsmith"
    },
    {
      "type":"twitter",
      "id":"johnsmith"
    }
  ]
}
```

5.3.1.4 Μεταδεδομένα εγγράφου (Document Metadata)

Ένα έγγραφο δεν αποτελείται μόνο από τα δεδομένα του. Έχει επίσης πληροφορίες μεταδεδομένων σχετικά με το έγγραφο. Τα τρία απαιτούμενα στοιχεία μεταδεδομένων έχουν ως εξής:

- Ευρετήριο(index): Το οποίο προσδιορίζει το που βρίσκεται το έγγραφο
- Type: Η κλάση του αντικειμένου που αντιπροσωπεύει το έγγραφο
- Id: Το μοναδικό αναγνωριστικό για το έγγραφο

_index

Ένας index είναι σαν μια βάση δεδομένων σε μια σχεσιακή βάση δεδομένων. Είναι ο τόπος που αποθηκεύουμε και δημιουργούμε ευρετήριο των σχετικών δεδομένων.

Για τη σωστή ευρετηρίαση το μόνο που έχουμε να κάνουμε είναι να επιλέξουμε ένα όνομα για τον index. Αυτό το όνομα πρέπει να είναι πεζά, δεν μπορεί να ξεκινήσει με μια υπογράμμιση και δεν μπορεί να περιέχει κόμματα. Θα μπορούσαμε με απλά λόγια να χρησιμοποιήσουμε έναν ιστότοπο ως όνομα ευρετηρίου για τα περιεχόμενα του.

_type

Στις εφαρμογές, χρησιμοποιούμε αντικείμενα που αντιπροσωπεύουν πράγματα όπως ένας χρήστης, μια ανάρτηση ιστολογίου, ένα σχόλιο ή ένα μήνυμα ηλεκτρονικού ταχυδρομείου.

Κάθε αντικείμενο ανήκει σε μια κλάση που καθορίζει τις ιδιότητες ή τα δεδομένα που σχετίζονται με ένα αντικείμενο. Τα αντικείμενα στην κατηγορία χρηστών μπορεί να έχουν όνομα, φύλο, ηλικία και διεύθυνση ηλεκτρονικού ταχυδρομείου.

Σε μια σχεσιακή βάση δεδομένων, συνήθως αποθηκεύουμε αντικείμενα της ίδιας κλάσης στον ίδιο πίνακα, επειδή μοιράζονται την ίδια δομή δεδομένων. Για τον ίδιο λόγο, στο Elasticsearch χρησιμοποιούμε τον ίδιο τύπο για έγγραφα που αντιπροσωπεύουν την ίδια κλάση, επειδή μοιράζονται την ίδια δομή δεδομένων.

Κάθε τύπος έχει τον δικό του ορισμό χαρτογράφησης ή σχήματος (mappings), ο οποίος καθορίζει τη δομή δεδομένων για έγγραφα αυτού του τύπου, όπως και οι στήλες σε έναν πίνακα βάσης δεδομένων. Έγγραφα όλων των τύπων μπορούν να αποθηκευτούν στο ίδιο ευρετήριο, αλλά η χαρτογράφηση για κάθε τύπο υποδηλώνει στο Elasticsearch πώς θα πρέπει να αναπροσαρμόζονται τα δεδομένα σε κάθε έγγραφο.

Ένα όνομα _ type μπορεί να είναι πεζά ή κεφαλαία, αλλά δεν πρέπει να ξεκινά με υπογράμμιση ή να περιέχει κόμματα. Θα χρησιμοποιήσουμε το blog για το όνομα του τύπου μας.

_id

Το αναγνωριστικό ID είναι μια συμβολοσειρά που, όταν συνδυάζεται με το _index και _type, αναγνωρίζει μοναδικά ένα έγγραφο στο Elasticsearch. Όταν δημιουργείται ένα νέο έγγραφο, μπορεί είτε να δοθεί το δικό μας _id είτε να αφήσουμε το Elasticsearch να δημιουργήσει ένα για εμάς.

Λειτουργίες πάνω στα έγγραφα

Η ενέργεια που εκτελούνται στα έγγραφα είναι οι ακόλουθες:

• **Create.** Δημιούργησε ένα έγγραφο μόνο εάν το έγγραφο δεν υπάρχει ήδη.

• **Index.** Δημιούργησε ένα νέο έγγραφο ή αντικαταστήστε ένα υπάρχον έγγραφο.

• **Update**. Κάντε μερική ενημέρωση σε ένα έγγραφο.

• **Delete**. Διαγραφή εγγράφου.

Τα μεταδεδομένα καθορίζουν _index, _type και _id του εγγράφου που θα ευρετηριάζονται, δημιουργούνται, ενημερώνονται ή διαγράφονται.

Για παράδειγμα, ένα αίτημα διαγραφής μπορεί να μοιάζει με αυτό:

{ "delete": { "_index": "website", "_type": "blog", "_id": "123" } }

Το αίτημα προς στο σύστημα αποτελείται από το ίδιο το έγγραφο πηγή - τα πεδία και τις τιμές που περιέχει το έγγραφο. Τέτοια αιτήματα αποστέλλονται για τη ευρετηρίαση και τη δημιουργία, ώστε να γίνει δυνατή η εισαγωγή του εγγράφου στον index.

Παρόμοια αιτήματα αποστέλλονται προς το API ενημέρωσης για λειτουργίες ενημέρωσης: doc, upsert, script και ούτω καθεξής. Δεν απαιτείται γραμμή σώματος στο αίτημα που αποστέλλεται για διαγραφή.

{ "create": { "_index": "website", "_type": "blog", "_id": "123" }}

{ "title": "My first blog post" }

Εάν δεν έχει οριστεί _id, ένα ID θα δημιουργηθεί αυτόματα:

{ "index": { "_index": "website", "_type": "blog" }}

{ "title": "My second blog post" }

5.3.2 Βασικές αρχές αναζήτησης Elasticsearch

Το Elasticsearch όχι μόνο αποθηκεύει το έγγραφο, αλλά και ευρετηριάζει το περιεχομένου του και το καθιστά έτοιμο για αναζήτηση.

Κάθε πεδίο σε ένα έγγραφο ευρετηριάζεται και μπορεί να ανακτηθεί μέσω αναζήτησης. Κατά τη διάρκεια ενός μόνο ερωτήματος, το Elasticsearch μπορεί να χρησιμοποιήσει όλους αυτούς τους κόμβους, για να επιστρέψει τα αποτελέσματα σε εκπληκτική ταχύτητα. Αυτό είναι κάτι που ποτέ δεν θα μπορούσε να γίνει σε μια παραδοσιακή βάση δεδομένων.

Μια αναζήτηση μπορεί να είναι οποιαδήποτε από τις παρακάτω:

- Ένα **καθορισμένο ερώτημα** σε συγκεκριμένα πεδία όπως το φύλο ή η ηλικία, ταξινομημένα σε κάποιο πεδίο όπως το join_date, παρόμοιο με τον τύπο ερωτήματος που θα μπορούσε να κατασκευαστεί σε SQL
- Ένα **ερώτημα πλήρους κειμένου**, το οποίο βρίσκει όλα τα έγγραφα που ταιριάζουν με τις λέξεις-κλειδιά αναζήτησης και τα επιστρέφει ταξινομημένα κατά συνάφεια
- Ένας **συνδυασμός** των δύο

Πολλές αναζητήσεις μπορούν να λειτουργήσουν χωρίς ιδιαίτερη παραμετροποίηση του Elasticsearch, για να χρησιμοποιηθεί όμως στο πλήρες δυναμικό του, θα πρέπει να κατανοήσουμε τρία θέματα:

- **Mapping**. Πώς ερμηνεύονται τα δεδομένα σε κάθε πεδίο
- **Analysis**. Πώς επεξεργάζεται το πλήρες κείμενο για γίνει η αναζήτηση
- **Query DSL**. Η ευέλικτη, ισχυρή γλώσσα ερωτήσεων που χρησιμοποιείται από το Elasticsearch

5.3.2.1 Mapping

Για να μπορέσουμε να επεξεργαστούμε τα πεδία ημερομηνίας ως ημερομηνίες, αριθμητικά πεδία ως αριθμούς και πεδία συμβολοσειρών ως συμβολοσειρές πλήρους κειμένου ή ακριβούς

τιμής, το Elasticsearch πρέπει να γνωρίζει τον τύπο δεδομένων που περιέχει κάθε πεδίο. Αυτές οι πληροφορίες περιέχονται στο Mapping. [55]

Όπως εξηγείται κάθε έγγραφο σε έναν index έχει έναν τύπο. Κάθε τύπος έχει τη δική του Mapping. Ένα Mapping ορίζει τα πεδία μέσα σε έναν τύπο, τον τύπο δεδομένων για κάθε πεδίο και τον τρόπο χειρισμού του πεδίου από το Elasticsearch. Ένα Mapping χρησιμοποιείται επίσης για τη διαμόρφωση των μεταδεδομένων που σχετίζονται με τον τύπο.

Οι βασικοί τύποι πεδίου του Elasticsearch υποστηρίζουν τους ακόλουθους απλούς τύπους πεδίων:

- String: συμβολοσειρά
- Whole number: byte, short, integer, long
- Floating point: float, double
- Boolean: boolean
- Date: ημερομηνία

Όταν εκτελείται η ευρετηρίαση ενός εγγράφου που περιέχει ένα νέο πεδίο το οποίο προηγουμένως δεν εμφανιζόταν το Elasticsearch θα χρησιμοποιηθεί το δυναμική Mapping για να προσπαθήσει να μαντέψει τον τύπο πεδίου από τους βασικούς τύπους δεδομένων που είναι διαθέσιμοι στο JSON, χρησιμοποιώντας τους ακόλουθους κανόνες:

- **JSON type**

  Field type
- **Boolean**: true or false

  boolean
- **Whole number**: 123

  long
- **Floating point**: 123.45

  double
- **String**, **valid date**: 2014-09-15

  date
- **String**: foo bar

  string

5.3.2.2 Αναλυτές (Analyzers)

Ενώ οι βασικοί τύποι δεδομένων επαρκούν για πολλές περιπτώσεις, θα πρέπει συχνά να προσαρμόσουμε στο mapping μεμονωμένων πεδίων, και πιο συγκεκριμένα σε ειδικά πεδία συμβολοσειρών. Οι προσαρμοσμένες αντιστοιχίσεις μας επιτρέπουν να κάνουμε τα εξής:

- Διαχωρισμό των πεδίων συμβολοσειρών πλήρους κειμένου και των πεδίων συμβολοσειρών ακριβούς τιμής
- Χρήση αναλυτών ειδικής γλώσσας
- Βελτιστοποίηση ενός πεδίου για μερική αντιστοίχιση
- Καθορισμό προσαρμοσμένων μορφών ημερομηνίας

Ο index ελέγχει τον τρόπο με τον οποίο η συμβολοσειρά θα ευρετηριάζεται. Μπορεί να περιέχει μία από τις τρεις τιμές:

- **analyzed.** Πρώτα αναλύστε τη συμβολοσειρά και στη συνέχεια ευρετηρίασε την. Με άλλα λόγια, ευρετηρίασε αυτό το πεδίο ως πλήρες κείμενο.
- **not_analyzed.** Ευρετηρίασε αυτό το πεδίο, έτσι ώστε να μπορεί να αναζητηθεί, αλλά ευρετηρίασε την τιμή ακριβώς όπως καθορίζεται. Να μην αναλυθεί.
- **no.** Να μην ευρετηριαστεί καθόλου αυτό το πεδίο. Αυτό το πεδίο δεν θα μπορεί να αναζητηθεί.

Η προεπιλεγμένη τιμή του index για ένα πεδίο συμβολοσειράς είναι analyzed. Εάν θέλουμε να ευρετηριαστεί το πεδίο ως ακριβή τιμή, πρέπει να το ορίσουμε σε not_analyzed:

```
{
  "tag":{
    "type":"string",
    "index":"not_analyzed"
  }
}
```

Αναλυτές γλώσσας (Language Analyzers)

Ενώ οι αναλυτές γλωσσών μπορούν να χρησιμοποιηθούν χωρίς κάποια ιδιαίτερη διαμόρφωση, οι περισσότεροι από αυτούς μας επιτρέπουν να ελέγχουμε πτυχές της συμπεριφοράς τους και πιο συγκεκριμένα:

Εξαίρεση λέξεων-κλειδιών (Stem-word exclusion)

Για παράδειγμα, οι χρήστες που αναζητούν τον «World Health Organization» αντλούν αντ' αυτού αποτελέσματα για την «organ health». Ο λόγος για αυτή τη σύγχυση είναι ότι τόσο το «organ» όσο και η «organization» προέρχονται από την ίδια λέξη: organ. Συχνά αυτό δεν είναι πρόβλημα, αλλά σε αυτή τη συλλογή εγγράφων συγκεκριμένου γνωστικού τομέα, αυτό οδηγεί σε συγκεχυμένα αποτελέσματα. Θα θέλαμε να αποτρέψουμε το stemmed των λέξεων organization και organizations.

Προσαρμοσμένα stopwords

Η προεπιλεγμένη λίστα stopwords που χρησιμοποιούνται στα αγγλικά είναι η εξής:

a, an, and, are, as, at, be, but, by, for, if, in, into, is, it, no, not, of, on, or, such, that, the, their, then, there, these, they, this, to, was, will, with. Το ασυνήθιστο για το no και not είναι ότι αντιστρέφουν την έννοια των λέξεων που τους ακολουθούν. Υπάρχουν περιπτώσεις όπου αυτές οι δύο λέξεις είναι σημαντικές και ότι δεν πρέπει να τις αντιμετωπίζουμε ως stopwords.

Προκειμένου να προσαρμόσουμε τη συμπεριφορά του english analyzer, πρέπει να δημιουργήσουμε έναν custom analyzer που χρησιμοποιεί τον english analyzer ως βάση του, αλλά προσθέτει κάποια διαμόρφωση:

```
PUT /my_index{
  "settings":{
    "analysis":{
      "analyzer":{
        "my_english":{
          "type":"english",
          "stem_exclusion":[
            "organization",
            "organizations"
          ],
          "stopwords":[
            "a",  "an", "and", are",..... "these", "they", "this", "to", "was", "will",  "with"
          ] }}}}}
```

5.3.2.3 Δομημένη αναζήτηση (Structured Search)

Η δομημένη αναζήτηση αφορά την ενασχόληση με δεδομένα που έχουν εγγενή δομή. Οι ημερομηνίες, οι χρόνοι και οι αριθμοί είναι όλα δομημένα: έχουν μια ακριβή μορφή στην οποία μπορούμε να εκτελέσουμε λογικές λειτουργίες. Οι κοινές λειτουργίες περιλαμβάνουν τη σύγκριση εύρους αριθμών ή ημερομηνιών ή τον προσδιορισμό των δύο τιμών που είναι μεγαλύτερες.

Με τη δομημένη αναζήτηση, η απάντηση στην ερώτησή είναι πάντα ναι ή όχι. κάτι είτε ανήκει ή δεν ανήκει στο σετ. Η δομημένη αναζήτηση δεν ασχολείται για τη συνάφεια ή τη βαθμολόγηση του εγγράφου. Απλά περιλαμβάνει ή αποκλείει τα έγγραφα.

Ένας αριθμός δεν μπορεί να είναι περισσότερο σε ένα εύρος από οποιοδήποτε άλλο αριθμό που εμπίπτει στο ίδιο εύρος. Είτε είναι στο εύρος είτε δεν είναι. Ομοίως, για δομημένο κείμενο, μια τιμή είναι είτε ισότιμη είτε όχι.

Ο όρος φίλτρο δεν είναι πολύ χρήσιμος μόνος του. Όπως αναφέρθηκε, το API αναζήτησης αναμένει ένα ερώτημα και όχι ένα φίλτρο. Για να χρησιμοποιήσουμε το φίλτρο όρων μας, πρέπει να το τυλίξουμε με φιλτραρισμένο ερώτημα:

```
GET /my_store/products/_search{
  "query":{
    "filtered":{
      "query":{
        "match_all":{

        }
      },
      "filter":{
        "term":{
          "price":20
        }
      }
    }
  }
}
```

5.3.2.4 Αναζήτηση πλήρους κειμένου (Full-Text Search)

Μετά την απλή περίπτωση αναζήτησης δομημένων δεδομένων, θα μελετήσουμε την αναζήτηση πλήρους κειμένου: πώς να αναζητήσουμε μέσα σε πεδία πλήρους κειμένου για να βρούμε τα πιο σχετικά έγγραφα.

Οι δύο πιο σημαντικές πτυχές της αναζήτησης πλήρους κειμένου είναι οι εξής:

- **Συνάφεια.** Η ικανότητα ταξινόμησης των αποτελεσμάτων ανάλογα με το κατά πόσο συναφή είναι με το δεδομένο ερώτημα. Η συνάφεια υπολογίζεται χρησιμοποιώντας την TF / IDF, εγγύτητα σε μια συγκεκριμένη θέση, ασαφή ομοιότητα ή κάποιον άλλο αλγόριθμο.
- **Ανάλυση.** Η διαδικασία μετατροπής ενός μπλοκ κειμένου σε ξεχωριστά, κανονικοποιημένα tokens για (α) τη δημιουργία ενός ανεστραμμένου index και (β) την αναζήτηση του ανεστραμμένου index .

Όταν μιλάμε είτε για τη συνάφεια είτε για την ανάλυση, βρισκόμαστε στο πεδίο των ερωτημάτων και όχι των φίλτρων.

Αναζήτηση βασισμένη σε όρους έναντι αναζήτησης πλήρους κειμένου

Ενώ όλα τα ερωτήματα ενέχουν κάποιο υπολογισμό συνάφειας, δεν έχουν φάση ανάλυσης. Εκτός από εξειδικευμένα ερωτήματα όπως τα ερωτήματα bool ή function_score, τα οποία δεν λειτουργούν με κείμενο. Τα ερωτήματα κειμένου μπορούν να χωριστούν σε δύο οικογένειες:

Ερωτήματα που βασίζονται σε όρους. Η αναζήτηση όρων ή τα ασαφή ερωτήματα είναι ερωτήματα χαμηλού επιπέδου που δεν έχουν φάση ανάλυσης. Λειτουργούν με ένα μόνο όρο. Το ερώτημα για τον όρο Foo αναζητά αυτόν τον ακριβή όρο στον ανεστραμμένο δείκτη και υπολογίζει το TF / IDF relevance_score για κάθε έγγραφο που περιέχει τον όρο.

Είναι σημαντικό να θυμόμαστε ότι ο όρος ερώτημα εμφανίζεται στον ανεστραμμένο δείκτη μόνο για τον ακριβή όρο. Δεν θα ταιριάζει με παραλλαγές όπως foo ή FOO. Δεν έχει σημασία πως ο όρος βρέθηκε στο ευρετήριο, αλλά το ότι είναι μέσα.

Σε μια αναζήτηση όρων που περιλαμβάνει τους όρους ["Foo","Bar"] με ακριβείς τιμές σε πεδίο not_analyzed, ή το Foo Bar σε ένα πεδίο ανάλυσης με τον αναλυτή κενών διαστημάτων, θα προέκυπταν οι δύο όροι Foo και Bar στον ανεστραμμένο δείκτη.

Ερωτήματα πλήρους κειμένου Αιτήματα

Τα ερωτήματα αντιστοίχισης ή query_string είναι ερωτήματα υψηλού επιπέδου που κατανοούν τα πεδία ενός mapping:

- Αν τα χρησιμοποιήθούν για να υποβάλουμε ερώτημα σε ένα πεδίο ημερομηνίας ή ακέραιου, θα αντιμετωπίσουν τη συμβολοσειρά ερωτήματος ως ημερομηνία ή ακέραιος, αντίστοιχα.
- Εάν ερωτήσουμε ένα πεδίο συμβολοσειράς ακριβούς τιμής (not_analyzed), θα αντιμετωπίσουν ολόκληρη τη συμβολοσειρά του ερωτήματος ως έναν μόνο όρο.
- Αν όμως κάνουμε ένα ερώτημα σε πεδίο πλήρους κειμένου, θα περάσουν πρώτα τη συμβολοσειρά ερωτήματος, μέσω του κατάλληλου αναλυτή, για να παράγουν τη λίστα των όρων που θα χρησιμοποιηθούν σε αυτό.

Μόλις το ερώτημα συγκεντρώσει μια λίστα όρων, εκτελεί το κατάλληλο ερώτημα χαμηλού επιπέδου για κάθε έναν από αυτούς τους όρους και στη συνέχεια συνδυάζει τα αποτελέσματά τους για να παράγει την τελική βαθμολογία συνάφειας για κάθε έγγραφο.

Σπανίως πρέπει να χρησιμοποιούνται άμεσα τα ερωτήματα που βασίζονται σε όρους. Συνήθως θέλουμε να αναζητήσουμε πλήρες κείμενο, όχι μεμονωμένους όρους, και αυτό είναι πιο εύκολο να το κάνουμε με τα ερωτήματα πλήρους κειμένου υψηλού επιπέδου (τα οποία καταλήγουν εσωτερικά σε ερωτήματα που βασίζονται σε όρους).

5.4 Προαπαιτούμενα για την υλοποίηση της εφαρμογής

Java 8

Η Java είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού που σχεδιάστηκε από την εταιρεία πληροφορικής Sun Microsyst. Java 8 και κατά την παρούσα φάση τρέχει στην (LTS) έκδοση της. [56]

Elasticsearch 2.4.2

Το Elasticsearch είναι μια μηχανή αναζήτησης βασισμένη στο Lucene. Παρέχει μια κατανεμημένη μηχανή αναζήτησης πλήρους κειμένου, με δυνατότητες πολλαπλών λειτουργιών, με διεπαφή ιστού HTTP και έγγραφα που αναπαρίστανται σε schema-free JSON μορφή. Το Elasticsearch αναπτύσσεται σε Java και κυκλοφορεί ως λειτουργικό ανοικτού κώδικα σύμφωνα με τους όρους της Άδειας Apache. Το Elasticsearch αναπτύσσεται παράλληλα με μια μηχανή συλλογής και καταγραφής δεδομένων που ονομάζεται Logstash και μια πλατφόρμα ανάλυσης και απεικόνισης που ονομάζεται Kibana. Τα τρία προϊόντα έχουν σχεδιαστεί για χρήση ως ολοκληρωμένη λύση, που αναφέρεται ως " Elastic Stack " (πρώην " ELK stack ").

Elasticsearch-head

Το Elasticsearch-head είναι μια διαδικτυακή διεπαφή για περιήγηση και αλληλεπίδραση με ένα cluster Elasticsearch.

PHP 7

Η PHP (PHP: Hypertext Preprocessor) είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που είτε θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML ή θα επεξεργασθεί τις εισόδους δίχως να προβάλλει την έξοδο στο χρήστη, αλλά θα τις μεταβιβάσει σε κάποιο άλλο PHP script. [57]

Apache 2

Ο Apache HTTP γνωστός και απλά σαν Apache είναι ένας εξυπηρετητής του παγκόσμιου ιστού (web). Όποτε ένας χρήστης επισκέπτεται ένα ιστότοπο το πρόγραμμα πλοήγησης (browser) επικοινωνεί με έναν διακομιστή (server) μέσω του πρωτοκόλλου HTTP, ο οποίος παράγει τις ιστοσελίδες και τις αποστέλλει στο πρόγραμμα πλοήγησης. Ο Apache λειτουργεί σε διάφορες πλατφόρμες όπως τα Windows, το Linux, το Unix και το Mac OS X. Κυκλοφορεί υπό την άδεια λογισμικού Apache και είναι λογισμικό ανοιχτού κώδικα. Συντηρείται από μια κοινότητα ανοικτού κώδικα με επιτήρηση από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation). [58]

php-xml

Αυτό το πακέτο παρέχει τις λειτουργικές μονάδες DOM, SimpleXML, WDDX, XML και XSL για την PHP. Πρόκειται για μια επέκταση που μας επιτρέπει να χειριζόμαστε εύκολα και να λαμβάνουμε δεδομένα XML.

modphp

Το modphp είναι ένας διερμηνέας PHP ενσωματωμένος μέσα στη διαδικασία Apache. Αυτή η προσέγγιση βοηθά το Apache και την PHP να επικοινωνούν καλύτερα.

Firebase php-jwt, έκδοση v4.0.0

Μια απλή βιβλιοθήκη για την κωδικοποίηση και αποκωδικοποίηση JSON Web Tokens (JWT) στην PHP, σύμφωνα με το RFC 7519.

Guzzle, PHP HTTP client

Το Guzzle είναι ένα πρόγραμμα-πελάτης HTTP PHP που διευκολύνει την αποστολή αιτημάτων HTTP.

Αποτελεί μια απλή διεπαφή για τη δημιουργία ερωτημάτων, αιτήσεων POST, ροής μεγάλων λήψεων, χρησιμοποιώντας cookies HTTP, μεταφόρτωσης δεδομένων JSON.

Μπορεί να στείλει συγχρονισμένα και ασύγχρονα αιτήματα χρησιμοποιώντας την ίδια διεπαφή.

Χρησιμοποιεί διασυνδέσεις τύπου PSR-7 για αιτήματα, απαντήσεις και ροές. Αυτό μας επιτρέπει να χρησιμοποιήσουμε άλλες συμβατές βιβλιοθήκες PSR-7 με το Guzzle.

Περιέχει λειτουργίες που μας βοηθούν να απομακρυνθούμε από το χαμηλό επίπεδο διαχείρισης του HTTP, επιτρέποντάς μας να γράψουμε κώδικα σε υψηλότερο επίπεδο, χωρίς να εξαρτόμαστε απόλυτα από το cURL PHP streams, sockets, ή non-bloc.

google-auth-library-php

Αυτή είναι η επίσημα υποστηριζόμενη, από την google βιβλιοθήκη PHP client για εξουσιοδότηση και έλεγχο ταυτότητας. Ο Auth 2.0 με τα API της Google παρέχει την εφαρμογή των προεπιλεγμένων διαπιστευτηρίων εφαρμογής για την PHP. Τα διαπιστευτήρια εφαρμογής παρέχουν έναν απλό τρόπο για να λάβουμε διαπιστευτήρια εξουσιοδότησης για χρήση στην κλήση API Google.

Google Cloud PHP Client

PHP client για υπηρεσίες Google Cloud Platform. Ο client υποστηρίζει τις υπηρεσίες του Google Cloud Platform και την Google Cloud Natural Language.

Bootstrap

HTML, CSS και JavaScript framework για την ανάπτυξη εφαρμογών για κινητές συσκευές για χρήση στο διαδίκτυο.

5.5 Εξαγωγή γεγονότων, οντοτήτων και σχέσεων (Micro Understanding)

Το γνωστικό πεδίο του micro-understanding αναφέρεται στην εξαγωγή μεμονωμένων οντοτήτων, γεγονότων ή σχέσεων από το κείμενο. Αυτό είναι χρήσιμο για:

- Εξαγωγή ακρωνυμίων και των ορισμών τους
- Εξαγωγή αναφορών παραπομπής σε άλλα έγγραφα
- Εξαγωγή οντοτήτων-κλειδιών (άτομα, εταιρεία, προϊόν, ποσά δολαρίου, τοποθεσίες, ημερομηνίες). Σημειώστε ότι η εξαγωγή οντοτήτων "κλειδιών" δεν είναι η ίδια με την εξαγωγή οντοτήτων "όλων" (υπάρχει κάποια διάκριση που προκύπτει από την επιλογή της οντότητας που είναι "κλειδί")
- Εξαγωγή γεγονότων και μεταδεδομένων από το πλήρες κείμενο.
- Εξαγωγή οντοτήτων με συναίσθημα (π.χ. θετικό συναίσθημα προς ένα προϊόν ή εταιρεία)
- Προσδιορισμός σχέσεων όπως επιχειρηματικές σχέσεις, στόχος / δράση / δράστης κλπ.
- Προσδιορισμός παραβιάσεων συμμόρφωσης, δηλώσεις που δείχνουν πιθανή παραβίαση κανόνων
- Εξαγωγή δηλώσεων με αναφορά, για παράδειγμα, αποσπάσματα από άτομα (ποιοι είπαν τι)
- Εξαγωγή κανόνων ή απαιτήσεων, όπως όροι συμβολαίων, απαιτήσεις ρυθμιστικών αρχών κ.λπ.

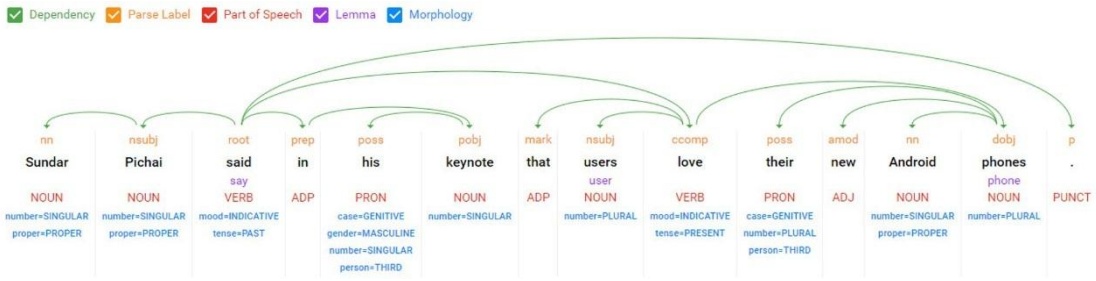
Το micro understanding πρέπει να γίνει με τη συντακτική ανάλυση του κειμένου. Αυτό σημαίνει ότι η σειρά και η χρήση των λέξεων είναι αρκετά σημαντική.

Υπάρχουν τρεις προσεγγίσεις για την πραγματοποίηση ανάλυσης micro understanding.

1.Top Down – Προσδιορίζει τα POS, κατανοεί και δημιουργεί διάγραμμα της φράσης σε συντακτική πρόταση, με ουσιαστικά, ρήματα, αντικείμενο και θέμα, επίθετα, επίρρημα κλπ., Μέσω αυτής της δομής προσδιορίζει τα σημεία ενδιαφέροντος.

- Πλεονεκτήματα - μπορεί να χειριστεί πολύπλοκες δομές και σχέδια τα οποία δεν έχει συναντήσει κατά το παρελθόν.
- Μειονεκτήματα – Δύσκολοι στην κατασκευή κανόνες, εύθραυστοι, ακόμη μπορεί να απαιτούν συμαντική επιπρόσθετη εργασία κατά της αντιστοίχιση προτύπων ακόμα και μετά την ανάλυση.

Δείγμα εξόδου top-down από το Google Cloud Natural Language API.



Εικόνα 14 Έξοδος top-down από το Google Cloud Natural Language API

Στο γράφημα βαθιάς κατανόησης, παρατηρούμε πώς όλοι οι modifiers συνδέονται μεταξύ τους. Παρατηρούμε επίσης, ότι απαιτείται ένα δεύτερο βήμα (το οποίο απαιτεί επιπρόσθετο προγραμματισμό) ώστε να προσδιοριστούν οι σχέσεις αντικειμένου / ενέργειας του γραφήματος που είναι κατάλληλες για εξαγωγή σε ένα γράφο ή σε δεδομένα κάποιας εφαρμογής.

2. Bottoms Up – Η μέθοδος αυτή δημιουργεί πολλά μοτίβα, ταιριάξει τα μοτίβα με το κείμενο και εξαγάγει τα απαραίτητα γεγονότα. Τα μοτίβα μπορούν να εισαχθούν με το χέρι ή μπορούν να υπολογιστούν με εξόρυξη κειμένου.

- Πλεονεκτήματα – Ευκολία δημιουργίας προτύπων, ενδείκνυται η υλοποίηση τους από επαγγελματίες χρήστες, δεν απαιτεί προγραμματισμό, ευκολία στον εντοπισμό λαθών και στην διόρθωση τους, γρήγορη εκτέλεση του αντίστοιχου κώδικα, αντιστοιχίζει άμεσα στις επιθυμητές εξόδους
- Μειονεκτήματα – Απαιτείται συνεχή συντήρηση των προτύπων, δεν μπορεί να αντιστοιχίσει με πρόσφατα αναγνωρισμένα μοτίβα.

3. Με χρήση στατιστικών – Μέθοδος παρόμοια με τα bottoms-up, αλλά ταιριάζει με τα πρότυπα σε σχέση με στατιστικά σταθμισμένα δεδομένα προτύπων που παράγονται από δεδομένα εκπαίδευσης.

- Πλεονεκτήματα – τα πρότυπα δημιουργούνται αυτόματα, ενσωματωμένοι στατιστικά συμβιβασμοί.
- Μειονεκτήματα – απαιτεί τη δημιουργία εκτεταμένων δεδομένων εκπαίδευσης (χιλιάδες παραδείγματα), θα πρέπει να επανεκπαιδεύονται περιοδικά τα μοντέλα για

καλύτερη ακρίβεια, δεν μπορεί να αντιστοιχίσει με πρόσφατα αναγνωρισμένα μοτίβα, δυσκολότερα στην διόρθωση.

Εικόνα 15 Παραδείγματα μοτίβων

Σημειώστε ότι αυτά τα μοτίβα μπορούν να εισαχθούν με το χέρι, ή μπορούν να παραχθούν στατιστικά (στατιστικά σταθμισμένα) με τη χρήση δεδομένων εκπαίδευσης ή να συνάγονται χρησιμοποιώντας τεχνικές εξόρυξης κειμένου και μηχανική μάθηση.

5.6 Ορθογραφικός έλεγχος

5.6.1 Βασικές έννοιες N-grams

Ένα 'N-gram' είναι μια ακολουθία N στοιχείων - γραμμάτων, λέξεων ή φωνημάτων. Γνωρίζουμε ότι ορισμένα ζεύγη (ή τριπλέτες, τετραπλέτες κ.λπ.) είναι πιθανό να εμφανιστούν πολύ πιο συχνά από άλλα. Για παράδειγμα, στις αγγλικές λέξεις, το U ακολουθεί πάντα το Q, και το αρχικό T δεν ακολουθείται ποτέ από το K (αν και μπορεί να συμβέι στην ουκρανική γλώσσα). Στα πορτογαλικά, ένα Ç ακολουθείται πάντα από ένα φωνήεν (εκτός από το E και το I). Δεδομένου ότι υπάρχουν επαρκή δεδομένα, μπορούμε να υπολογίσουμε τη συχνότητα κατανομής των δεδομένων για όλα τα N-γραμμάρια που εμφανίζονται σε αυτά τα δεδομένα. Επειδή οι μεταβολές αυξάνονται δραματικά με το N- gram παράδειγμα, το αγγλικό έχει 26 ^ 2 πιθανά ζεύγη χαρακτήρων, 26 ^ 3 τριπλέτες, και ούτω καθεξής. Το -N περιορίζεται σε ένα μέτριο αριθμό. Η Google έχει υπολογίσει τα δεδομένα λέξεων N-gram (N≤5) από τα δεδομένα ιστού, τα δεδομένα της διατίθενται ελεύθερα. [53]

Τα N-grams είναι ένα είδος μοντέλου Markov πολλαπλών τάξεων: η πιθανότητα ενός συγκεκριμένου στοιχείου στη θέση Nth εξαρτάται από τα προηγούμενα στοιχεία N-1 και

μπορεί να υπολογιστεί από δεδομένα. Μόλις υπολογιστεί, τα δεδομένα N-gram μπορούν να χρησιμοποιηθούν για διάφορους σκοπούς:

Μια προτεινόμενη εφαρμογή αυτόματης συμπλήρωσης λέξεων και φράσεων κατά την αναζήτηση, περιλαμβάνει:

- **Διόρθωση ορθογραφίας:** μια λέξη με λάθος γράμματα σε μια φράση μπορεί να επισημανθεί και να προταθεί η ορθή ορθογραφία με βάση τις σωστά ομιλούμενες γειτονικές λέξεις.
- **Αναγνώριση ομιλίας:** τα ομόφωνα (‘two’ vs ‘too’) μπορούν να αποσαφηνιστούν πιθανώς με βάση σωστά αναγνωρισμένες γειτονικές λέξεις.
- **Αποσαφήνιση λέξεων:** εάν κατασκευάσουμε N-grams για το νόημα των λέξεων όπου οι ομοιόμορφες λέξεις έχουν επισημανθεί, μπορούμε να χρησιμοποιήσουμε τις μη διφορούμενες γειτονικές λέξεις για να μαντέψουμε την ορθή φράση, αντιπαραβάλλοντας τη φράση με ορθά πρότυπα.

Τα δεδομένα N-gram είναι ογκώδη - η βάση δεδομένων N-gram της Google απαιτεί 28 GB - αλλά αυτό δεν είναι σημαντικό πρόβλημα καθώς η αποθήκευση γίνεται φθηνή. Ειδικές δομές δεδομένων, που ονομάζονται index N-gram, επιταχύνουν την αναζήτηση σε τέτοια δεδομένα. Οι ταξινομητές που βασίζονται σε N-gram χρησιμοποιούν το ακατέργαστο κείμενο εκπαίδευσης χωρίς σαφή γλωσσική γνώση / γνώση τομέα, παράγοντας καλές επιδόσεις, εννώ αφήνουν περιθώρια βελτίωσης σε περιπτώσεις που συνεργάζονται και με άλλες προσεγγίσεις.

5.6.2 Fuzzy Searches

Η αναζήτηση φυσικής γλώσσας είναι εγγενώς ασαφής. Δεδομένου ότι οι υπολογιστές δεν μπορούν να κατανοήσουν τη φυσική γλώσσα, υπάρχει μια πληθώρα προσεγγίσεων αναζήτησης, καθεμία με τα δικά της πλεονεκτήματα και μειονεκτήματα. Το Lucene, η τεχνολογία στην οποία βασίζεται το Elasticsearch, είναι ‘πολυεργαλείο’ που αποτελείται από πολλά εργαλεία επεξεργασίας κειμένου. Κάθε εργαλείο αποτελεί αλγοριθμική συντόμευση αντί της αληθινής γλωσσικής κατανόησης. [59]

Μερικά από αυτά τα εργαλεία, όπως ο Snowball stemmer και ο φωνητικός αναλυτής Metaphone, είναι πολύ εξελιγμένα. Αυτά τα εργαλεία μιμούνται γραμματικές και φωνητικές πτυχές της κατανόησης της γλώσσας. Άλλα εργαλεία είναι πολύ βασικά, όπως ο τύπος προθέματος του ερωτήματος (prefix query type), ο οποίος απλά ταιριάζει τα αρχικά γράμματα λέξεων. Τα ασαφή ερωτήματα κατατάσσονται, από την άποψη της πολυπλοκότητας, κάπου στη μέση αυτών των εργαλείων. Βρίσκουν λέξεις που χρειάζονται το πολύ ένα ορισμένο αριθμό αλλαγών χαρακτήρα, γνωστές ως "'edits'", για να ταιριάζουν με το ερώτημα. Για παράδειγμα, μια ασαφής αναζήτηση για το 'ax' θα ταιριάζει με τη λέξη 'axe', καθώς έχει μόνο μία διαγραφή, αφαιρώντας το «e», με τον τρόπο αυτό συσχετίζονται οι δύο λέξεις.

Τα ασαφή ερωτήματα μπορούν εύκολα να πραγματοποιηθούν μέσω πρόσθετων argument στον τύπο ερωτήματος αντιστοίχισης, όπως φαίνεται στο παρακάτω παράδειγμα αυτής της παραγράφου. Στο παράδειγμα, το τελικό αίτημα, σε μια αναζήτηση για το "vaacuum", που έχει ένα πρόσθετο A, πρέπει να εμφανίσει το προϊόν "Vacuum".

Το fuzziness argument προσδιορίζει τα αποτελέσματα ταιριάζουν με μέγιστη απόσταση επεξεργασίας 2. Πρέπει να σημειωθεί ότι η ασάφεια πρέπει να χρησιμοποιείται μόνο με τιμές 1 και 2, που σημαίνει ότι επιτρέπονται κατ' ανώτατο όριο 2 επεξεργασίες μεταξύ του ερωτήματος και ενός όρου σε ένα έγγραφο . Οι μεγαλύτερες διαφορές είναι πολύ πιο δαπανηρές υπολογιστικά δαπανηρές και δεν χρησιμοποιούνται από το Lucene.

```
{
 "query": {
  "match": {
   "name": {
    "query": "Vacuummm",
    "fuzziness": 2,
    "prefix_length": 1
```

5.6.3 Προσδιορισμός της απόστασης επεξεργασίας

Η μετρική που χρησιμοποιείται από τα ασαφή ερωτήματα για τον προσδιορισμό της ομοιότητας είναι η φόρμουλα απόστασης Damerau-Levenshtein. Με απλά λόγια, η απόσταση Damerau-Levenshtein μεταξύ δύο κομματιών κειμένου είναι ο αριθμός εισαγωγών, διαγραφών, αντικαταστάσεων και μεταθέσεων που απαιτούνται για να γίνει μια σειρά συμβολοσειρών όμοια με μια άλλη. Για παράδειγμα, η απόσταση Levenshtein μεταξύ των λέξεων "ax" και " axe " είναι 1 λόγω της απλής διαγραφής που απαιτείται.

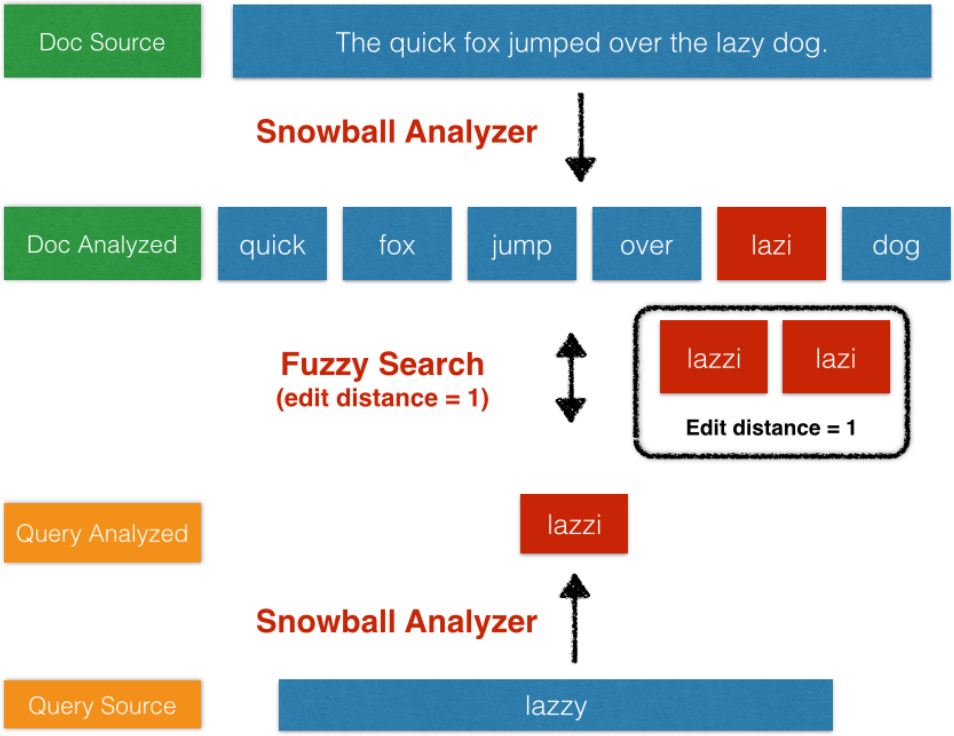
Η φόρμουλα αποστάσεων Damerau-Levenshtein είναι τροποποίηση της κλασικής απόστασης τύπου Levenshtein, με την προσθήκη της μεταφοράς ως έγκυρης λειτουργίας. Υποστηρίζονται και οι δύο τύποι, με το Damerau-Levenshtein να είναι η προεπιλογή και το κλασσικό Levenshtein να είναι επιλέξιμο, θέτοντας την ρύθμιση μεταθέσεων σε ψευδής στο ερώτημα. Η χρησιμότητα των μεταθέσεων μπορεί να φανεί στην περίπτωση της σύγκρισης των 'aex' και 'ax'.

Όταν χρησιμοποιούμε τον κλασικό τύπο απόστασης Levenshtein, το 'aex' δεν είναι ένα, αλλά δύο αλλαγές μακριά. το «e» πρέπει να διαγραφεί, μετά από το οποίο εισάγεται ένα νέο «e» στη σωστή θέση, ενώ στο Damerau-Levenshtein, αρκεί μια ενιαία πράξη, η οποία αντικαθιστά το «e» και το «x». Στην περίπτωση του κλασσικού Levenshtein θα σήμαινε ότι το «aex» είναι τόσο μακριά από το 'axe' όσο το 'faxes' είναι το οποίο μπορούμε να χρησιμοποιήσουμε σαν ένα

παράδειγμα που υποδεικνύει γιατί ο Damerau-Levenshtein έχει καλύτερη διαίσθηση στις περισσότερες περιπτώσεις.

Όταν ασχολούμαστε ιδιαίτερα με ασαφείς αναζητήσεις, είναι σημαντικό να καταλάβουμε ότι στο Elasticsearch το κείμενο τρέχει πρώτα μέσω ενός αναλυτή πριν γίνει διαθέσιμο για αναζήτηση. Όταν τα δεδομένα ευρετηριάζονται, μετατρέπονται σε αυτό που γνωρίζουμε σαν «όρους», τις πραγματικές μονάδες που μπορούν να αναζητηθούν στη βάση δεδομένων. Είναι οι όροι αυτοί που αναλύονται και όχι τα πραγματικά αποθηκευμένα έγγραφα που αναζητούνται. Αυτό σημαίνει ότι κατά την εκτέλεση ασαφών ερωτημάτων, το κείμενο του ερωτήματος μπορεί να συγκριθεί με μια μη αναμενόμενη τιμή ως αποτέλεσμα της ανάλυσης, οδηγώντας μερικές φορές σύγχυση των αποτελεσμάτων.

Αυτό σημαίνει επίσης ότι αν τα συνώνυμα είναι ενεργοποιημένα για ένα πεδίο, τα συνώνυμα μπορεί να ταιριάζουν, ακόμα κι αν η λέξη αυτή δεν εμφανίζεται καθόλου στο κείμενο προέλευσης. Για παράδειγμα, αν κάποιος χρησιμοποιήσει ένα ασαφές ερώτημα σε ένα πεδίο αναλύσεων N-gram, πιθανώς τα αποτελέσματα να είναι περίεργα, καθώς τα N-grams διαχωρίζουν τις λέξεις σε πολλούς μικρούς συνδυασμούς γραμμάτων, πολλά από τα οποία είναι μια ή δυο επεξεργασίες μακριά από το αρχικό, μπορεί οι πραγματικές λέξεις που συμμετέχουν είναι αρκετά διαφορετικές. Αυτό σημαίνει επίσης ότι εάν χρησιμοποιούμε έναν snowball analyzer, μια ασαφής αναζήτηση για τη λέξη 'running' θα έχει ως αποτέλεσμα σε μορφή περικοπής την λέξη 'run' αλλά δεν θα ταιριάζει με την λάθος σε ορθογραφία λέξη 'runninga' η οποία πηγάζει από το 'run' αλλά είναι περισσότερες από 2 αλλαγές μακριά από το 'runninga'. Στην περίπτωση αυτή μπορεί να προκληθεί αρκετή σύγχυση και γι' αυτό το λόγο, έχει νόημα να χρησιμοποιείται μόνο ο απλός αναλυτής, σε κείμενο προοριζόμενο για χρήση με ασαφή ερωτήματα, πιθανώς και η απενεργοποίηση των συνωνύμων. Για να διευκρινιστεί αυτή η περίπτωση, παρακάτω παρουσιάζουμε ένα διάγραμμα που δείχνει ένα ασαφές ερώτημα σε σχέση με ένα έγγραφο το οποίο αναλύεται με τη τεχνική snowball.



Εικόνα 16 Παράδειγμα ασαφής αναζήτησης

5.6.4 Οι διαφορετικοί τύποι ασαφών αναζητήσεων

Το Elasticsearch υποστηρίζει πολλαπλούς τύπους ασαφούς αναζήτησης, οι διαφορές αυτές μπορεί να προκαλέσουν σύγχυση. Η παρακάτω λίστα επιχειρεί να αποσαφηνίσει αυτούς τους τύπους.

- match query + fuzziness: Η προσθήκη της παραμέτρου ασάφειας σε ένα ερώτημα αντιστοίχισης μετατρέπει ένα ερώτημα απλής αντιστοίχισης σε ένα ασαφές. Αναλύει το κείμενο του ερωτήματος πριν εκτελέσει την αναζήτηση.
- fuzzy query: Ο τύπος αναζήτησης fuzzy query πρέπει γενικά να αποφεύγεται. Λειτουργεί ως ερώτημα με προκαθορισμένους όρους. Δεν αναλύει πρώτα το κείμενο του ερωτήματος.
- fuzzy_like_this / fuzzy_like_this_field: Ένα ερώτημα more_like_this, που υποστηρίζει ασάφεια, και έχει ένα βελτιστοποιημένο αλγόριθμο βαθμολόγησης που χειρίζεται καλύτερα τα χαρακτηριστικά των ασαφών αντιστοιχισμένων αποτελεσμάτων.
- suggesters: Οι suggesters δεν είναι ένας πραγματικός τύπος ερωτήματος, αλλά ένας ξεχωριστός τύπος λειτουργίας (εσωτερικά χτισμένος πάνω από ασαφείς ερωτήσεις) που μπορούν να εκτελεστούν είτε παράλληλα σε ένα ερώτημα είτε ανεξάρτητα. Οι προτάσεις είναι ιδανικές σε λειτουργίες τύπους 'Εννοείτε αυτό?'.

Ένα ερώτημα αντιστοίχισης με την παράμετρο ασάφειας είναι ίσως το πιο ευπροσάρμοστο από τα ασαφή ερωτήματα. Ο τύπος ασαφούς ερωτήματος υποστηρίζει την ίδια ακριβώς συμπεριφορά, εκτός από το ότι δεν επιτρέπει οποιαδήποτε ανάλυση για το κείμενο του ερωτήματος. Επιπλέον, ο τύπος ασαφούς ερωτήματος είναι ένα υποσύνολο της λειτουργικότητας ενός ερωτήματος αντιστοίχισης και το καθιστά πιο συγκεχυμένο από χρήσιμο.

Τα ερωτήματα fuzzy_like_this ή FLT είναι χρήσιμα για την παροχή συστάσεων βασισμένων σε ένα μεγάλο κομμάτι πηγαίου κειμένου που μπορεί να περιέχει ορθογραφικά λάθη ή άλλες ανακρίβειες, που διαχωρίζονται από την απόσταση επεξεργασίας. Υπάρχουν δύο διαφορετικά ερωτήματα σε αυτήν την κατηγορία fuzzy_like_this και fuzzy_like_this_field, τα οποία και τα δύο παρέχουν την ίδια λειτουργικότητα, ενώ το δεύτερο ερώτημα απλοποιεί τη σύνταξη για την περίπτωση όπου χρησιμοποιείται μόνο ένα πεδίο. Αυτά τα ερωτήματα παίρνουν μια παράμετρο, όπως_text, που αποτελείται από ένα μεγάλο κομμάτι κειμένου, πχ το σώμα ενός άρθρου, και προσπαθούν να βρουν έγγραφα όπως «αυτό». Το κείμενο στο άρθρο ελέγχεται και οι όροι ερωτήματος σταθμίζονται με βάση τη συχνότητά τους στο like_text με ειδικές τροποποιήσεις (απενεργοποίηση του συντελεστή συντονισμού, IDF βάσει πηγαίου κώδικα) για τον συνδυασμό των διαφόρων όρων από το κείμενο προέλευσης. Τα ερωτήματα FLT λειτουργούν καλύτερα για περιπτώσεις όπου το σώμα περιέχει μεγάλο αριθμό ορθογραφικών λαθών, διαφορετικά ένα τυπικό more_like_this ερώτημα θα έχει καλύτερη απόδοση.

Οι Suggesters είναι ιδανικοί στην περίπτωση που ένας χρήστης πληκτρολογεί το "New Yrok", για μια αναζήτηση, έχοντας σχεδιάσει να πληκτρολογήσει τη λέξη " New York ". Οι suggesters μπορούν να εμφανισθούν μέσω ενός κουτιού "εννοούσατε αυτό" στην διεπαφή χρήστη, το οποίο συνιστά μια ενδεχόμενη διόρθωση. Για τη συγκεκριμένη περίπτωση αναζήτησης τύπου πρόληψης πληκτρολόγησης, ένα σύστημα ολοκλήρωσης όρου προσφέρει καλύτερες επιδόσεις, δεδομένης της ευαισθησίας κατά την καθυστέρηση της εκτέλεσης της αναζήτησης.

5.7 Επέκταση ερωτημάτων και δημιουργία προτάσεων (query expansion and suggestion)

Η Ανάκτηση Πληροφοριών διαδραματίζει κεντρικό ρόλο στην εξερεύνηση και ερμηνεία δεδομένων εφαρμογών. Τα συστήματα ερωτήσεων βασισμένα σε λέξεις-κλειδιά είναι δημοφιλείς διεπαφές χρήστη. Οι χρησιμοποιούμενες φράσεις ερωτήσεων καθορίζουν την ποιότητα του αποτελέσματος αναζήτησης καθώς και την προσπάθεια που πρέπει να καταβάλει ένας χρήστης για την τελειοποίηση των επερωτήσεων.

Σε αυτό το πλαίσιο, η στοχευμένη βελτίωση των προτάσεων από τις μηχανές αναζήτησης αποτελεί από τις δύσκολες εργασίες των συστημάτων. Τα υπάρχοντα συστήματα υποστηρίζουν

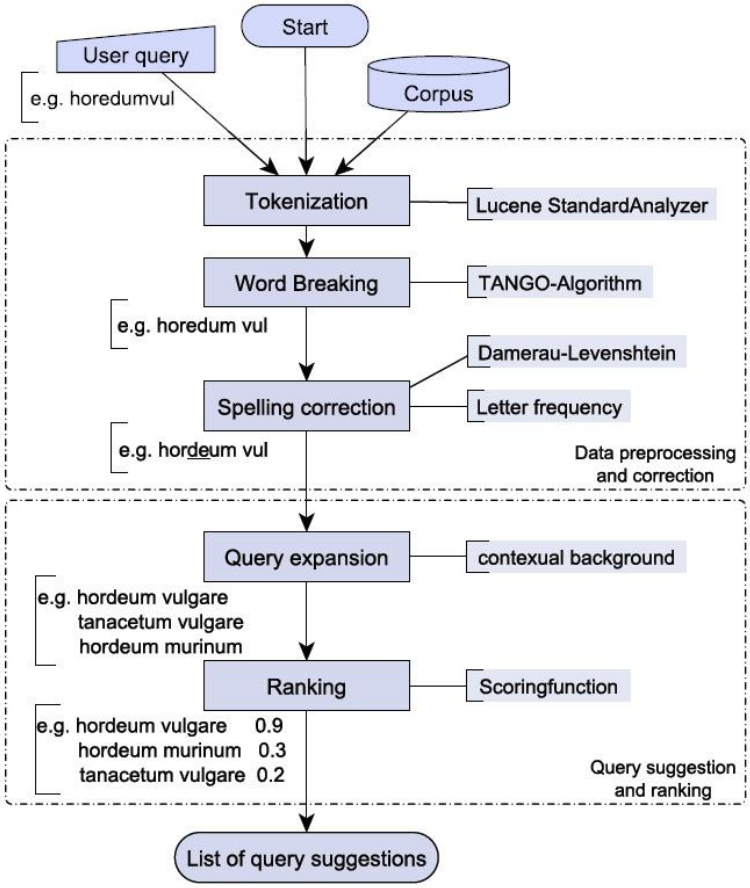
ερωτήματα όπως η διόρθωση ορθογραφίας, η βελτίωση επεξηγήσεων ή η επέκταση ερωτήματος. Ωστόσο, η πλειονότητα των front-end συστημάτων περιορίζει τη χρήση των βελτιωμένων αλγορίθμων ανάκτησης δεδομένων στο μέρος ερωτημάτων.

Η διόρθωση ορθογραφίας και η επέκταση ερωτήματος είναι δύο μέθοδοι οι οποίες παρέχουν διαισθητικά πρόσθετες πληροφορίες στους χρήστες ενώ πληκτρολογούν το ερώτημά τους. Αυτό μπορεί να οδηγήσει σε βελτιωμένη εμπειρία χρήστη και να αυξήσει την πιθανότητα ταυτοποίησης του ερωτήματος του χρήστη. Για τα περισσότερα ερωτήματα, υπάρχουν εκατοντάδες έως χιλιάδες έγγραφα που περιέχουν μερικούς ή όλους τους όρους στο ερώτημα. Μια μηχανή αναζήτησης πρέπει να ταξινομήσει τα έγγραφα με τον κατάλληλο τρόπο ώστε να δώσει στον χρήστη τις πιο σχετικές πληροφορίες. Η σημασία ενός εγγράφου σε σχέση με το ερώτημα του χρήστη αναφέρεται ως "συνάφεια εγγράφου". Συνήθως, αυτό είναι άγνωστο και πρέπει να εκτιμάται από τα χαρακτηριστικά του εγγράφου, του ερωτήματος, του ιστορικού χρήστη ή των στατιστικών μια εφαρμογής.

Ένας αντεστραμμένος index κειμένου που υλοποιείται για να υποστηρίζει την μέγιστη δυνατή αντιστοίχιση ασαφών tokens, στερεί τη δυνατότητα αλληλεπίδρασης με τον χρήστη ώστε το σύστημα να προτείνει καταλληλότερα ερωτήματα. Οι προτάσεις επεκτάσεως ερωτήματος παρέχουν έναν τρόπο να μειωθούν οι ψευδώς θετικές και οι ψευδώς αρνητικές εμφανίσεις και γίνονται πριν από την εκτέλεση ερωτήματος ώστε να αυξηθεί η χρηστικότητα της διεπαφής ερωτημάτων.

Χαρακτηριστικά συστημάτων δημιουργίας προτάσεων:

- Υποστηρίζουν ορολογία συγκεκριμένου πεδίου
- Διορθώνουν την ορθογραφία
- Επεκτείνουν τα ερωτήματα χωρίς ιστορικό καταγραφής χρηστών
- Παρέχουν προτάσεις αλληλοεπιδρώντας με τα ερωτήματα σε πραγματικό χρόνο



Εικόνα 17 Αρχιτεκτονική επέκτασης ερωτημάτων και δημιουργίας προτάσεων

5.7.1.1 Επέκταση ερωτημάτων

Η επέκταση ερωτημάτων είναι μια δύσκολη εργασία, για την οποία έχουν αναπτυχθεί προηγουμένως διαφορετικές μέθοδοι χρησιμοποιώντας αρχεία ερωτημάτων, λεξικά ή στατιστικές μέθοδοι για την επέκταση ενός ελλιπούς ερωτήματος. Μας ενδιαφέρει μια πρόταση ερώτησης που όχι μόνο συμπληρώνει την ατελή λέξη, αλλά και επεκτείνει το ερώτημα με επιπλέον ομάδες λέξεων που προέρχονται από το corpus του κειμένου. Μια μέθοδος για την υποβολή ερωτημάτων χωρίς τη χρήση ιστορικού δεν λαμβάνει υπόψη μόνον τα μοτίβα των tokens, αλλά και τις πλησιέστερες λέξεις. Το ερώτημα χρήστη στο παραπάνω παράδειγμα θα χωριστεί σε δύο μέρη, το ένα αφορά το νόημα του ερωτήματος και το δεύτερο μια τελευταία (υπο) λέξη. Στη συνέχεια, θα πραγματοποιηθούν δύο βήματα. Πρώτον, θα αναζητηθούν πιθανές ολοκληρώσεις στην τελευταία υπο-λέξη. Δεύτερον, δεύτερον θα προσπαθήσει να βρει πιθανές επεκτάσεις της υπο-λέξης.

5.7.1.2 Δημιουργία προτάσεων αναζήτησης

Μια άλλη προσέγγιση για την επίλυση της ασάφειας στα συστήματα ανάκτησης πληροφοριών είναι η δημιουργία προτάσεων αναζήτησης. Είναι πολύ συνηθισμένο για έναν χρήστη να αναδιατυπώνει το ερώτημά του όταν δεν έλαβε το ιδανικό αποτέλεσμα από το αρχικό ερώτημα. Το σύστημα μπορεί να βελτιώσει την αναζήτηση του χρήστη, παρέχοντας προτάσεις προσπαθώντας να προβλέψει την πρόθεση του χρήστη, σύμφωνα με τη συμπεριφορά του χρήστη στο παρελθόν ή μέσω άλλων μεθόδων. Μια σειρά πειραμάτων στην αλληλεπίδραση ανθρώπου-μηχανής του συστήματος ανάκτησης πληροφοριών δείχνουν ότι αντί της αυτόματης επέκτασης επερωτήσεων, οι χρήστες προτιμούν να χρησιμοποιούν τα προτεινόμενα ερωτήματα για να βελτιώσουν την αποτελεσματικότητα του αρχικού τους ερωτήματος. Η παροχή αποτελεσματικών και χρήσιμων προτάσεων είναι το κίνητρο για την δημιουργία προτάσεων αναζήτησης.

Προτάσεις αναζήτησης βάσει προηγούμενων επίλογων.

Η δημιουργία ερωτημάτων αναζήτησης βάσει επιλογών βασίζεται στην εξόρυξης των πρότυπων ενεργειών του χρήστη και σε ένα αρχείο καταγραφής αναζητήσεων. Εμφανίζονται ίχνη των ενεργειών για κάθε ερώτημα. Οι επισκέψεις σε προηγούμενες αναζητήσεις μπορεί να χρησιμοποιηθούν για την εκμετάλλευση της σχέσης μεταξύ διαφορετικών ερωτημάτων. Εάν τα ερωτήματα στο ίδιο σύμπλεγμα ταξινομούνται με το ίδιο ή παρόμοιο θέμα, τα ερωτήματα μέσα στο σύμπλεγμα θα χρησιμοποιηθούν ως προτάσεις αναζήτησης.

Σε ορισμένα συστήματα προτάσεων αντί να παρέχονται παρόμοιες προτάσεις σε κάθε χρήστη, συγκέντρωσαν δεδομένα προηγούμενων επισκέψεων για ορισμένους τύπους χρηστών για να προβλέψουν την πρόθεση και την προτίμηση τους. Μια εξατομικευμένη πρόταση αναζήτησης δόθηκε σε κάθε χρήστη με βάση την προηγούμενη συμπεριφορά του.

Προτάσεις αναζήτησης βασισμένες στο session

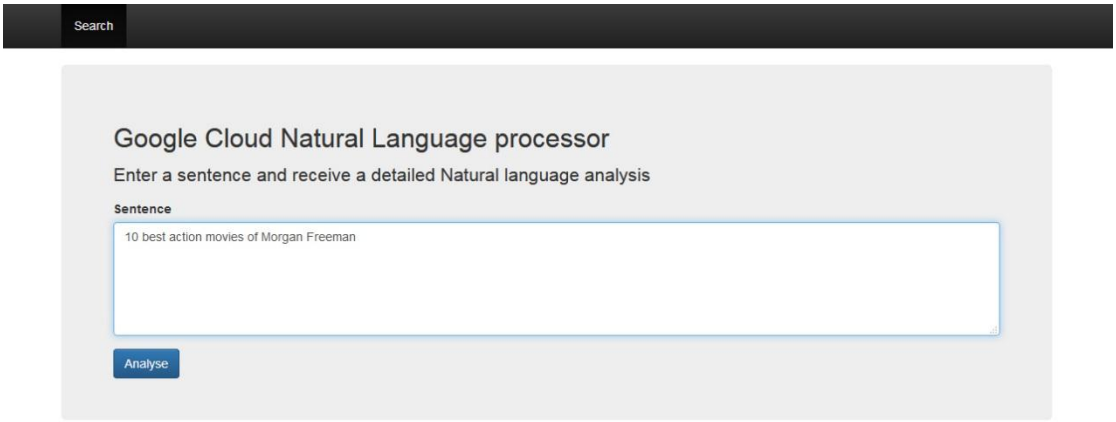
Οι προτάσεις αναζήτησης βασισμένες σε session στοιχειοθετούνται στην παραδοχή ότι σε κάθε ερώτημα αναζήτησης στην ίδια συνεδρία, συσχετίζονται με τον ένα ή τον άλλο τρόπο. Μερικές υποθέσεις μπορούν να γίνουν σχετικά με τις προτάσεις αναζήτησης βασισμένες σε session.

1. Όταν ένας αριθμός ερωτημάτων στην ίδια συνεδρία σε σύντομο χρονικό διάστημα συνήθως υποβάλλονται από τον ίδιο χρήστη.
2. Στην ίδια συνεδρία, ο χρήστης συχνά προσπάθησε να αλλάξει το ερώτημά του ή να δοκιμάσει ένα νέο ερώτημα, για να πάρει ένα καλύτερο αποτέλεσμα.
3. Τα ερωτήματα που υποβάλλονται από έναν χρήστη στην ίδια συνεδρία συνήθως αφορούν ένα μόνο θέμα.

5.8 Διεπαφή εφαρμογής

Στο παρακάτω τμήμα παραθέτουμε την διεπαφή της εφαρμογής. Η διεπαφή είναι διαδικτυακή, υλοποιημένη σε php html, με χρήση curl για την αποστολή των ερωτημάτων προς το Google NLP και προς το Elasticsearch. Χρησιμοποιεί Oauth για την αυθεντικοποίηση στο σύστημα του Google NLP. Ο εξυπηρετητής είναι ο Apache και γίνεται χρήση των πακέτων, php-xml, modphp, Firebase php-jwt, google-auth-library-php και Google Cloud PHP Client. Η εφαρμογή μπορεί να χρησιμοποιηθεί από φορητές συσκευές μέσω φυλλομετρητή.

Ακολουθεί εικόνα της κεντρικής σελίδα της εφαρμογής.



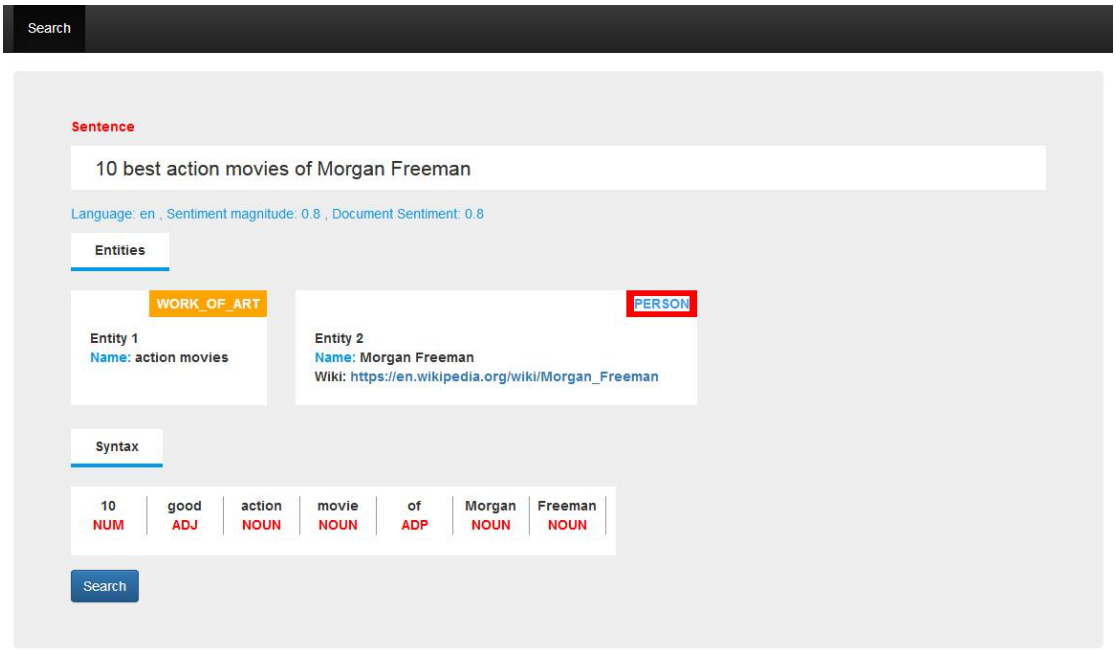
Εικόνα 18 Κεντρική σελίδα της εφαρμογής

Στην σελίδα αυτή ο χρήστης δίνει το ερώτημα σε μορφή ελεύθερου κειμένου. Στην εικόνα 18 παραθέτουμε το παράδειγμα του ερωτήματος '10 best action movies of Morgan Freeman'. Πατώντας στο κουμπί 'Analyze' το ερώτημα στέλνεται σε μορφή json στην διεπαφή του Google NLP και στην συνέχεια παραλαμβάνουμε το αποτέλεσμα στην ίδια μορφή. Στην συνέχεια η εφαρμογή αναλύει την απόκριση στο ερώτημα και εμφανίζει σε συγκεκριμένες περιοχές τα αποτελέσματα της ανάλυσης.

Στην εικόνα 19 βλέπουμε την ανάλυση του παραπάνω ερωτήματος. Αρχικά παρουσιάζεται το ερώτημα και από κάτω ακολουθούν τα αποτελέσματα της συναισθηματικής ανάλυσης. Βλέπουμε ότι τα επιμέρους αποτελέσματα είναι Sentiment magnitude: 0.8 , Document Sentiment: 0.8, λόγο του μικρού μήκους του ερωτήματος και της λέξης 'best', ότι είναι θετικό το συναίσθημα και η κλίμακα του σχετικά μεγάλη.

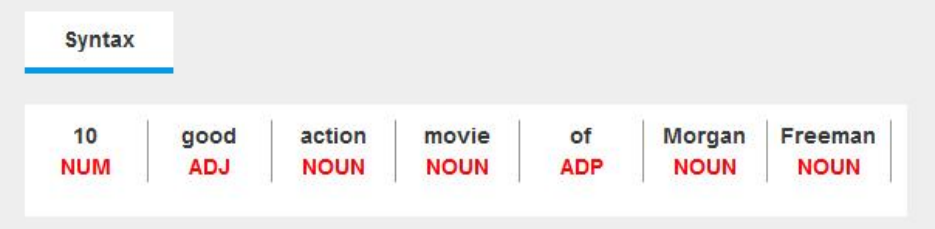
Στην συνέχεια έχουμε παρουσιάζονται οι οντότητες που εντοπίστηκαν στο ερώτημα και ο τύπος τους. Έχουμε τις οντότητες 'action movies' τύπου WORK_OF_ART και Morgan

Freeman τύπου PERSON, παραθέτετε και σύνδεσμος με την ιστοσελίδα του ηθοποιού στην Wikipedia.



Εικόνα 19 Ανάλυση ερωτήματος '10 best action movies of Morgan Freeman'

Στο επόμενο τμήμα βλέπουμε την συντακτική ανάλυση του ερωτήματος. Παρατηρούμε ότι στην ανάλυση αυτή το best έχει γίνει good και το movies movie, λόγο του lemmatization.

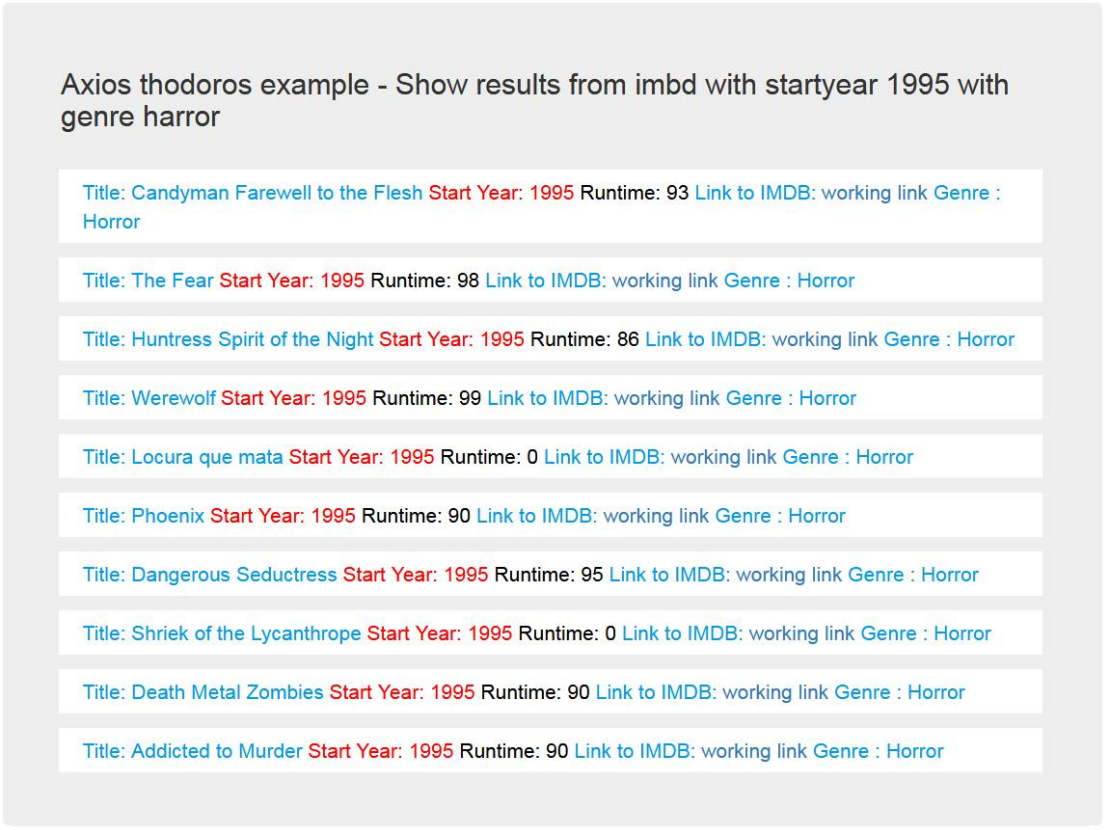


Εικόνα 20 Συντακτική ανάλυση ερωτήματος

Πατώντας search μεταφερόμαστε στην σελίδα των αποτελεσμάτων. Κατά την φάση αυτή γίνεται χρήση των δεδομένων της ανάλυσης του Google NLP, ώστε να αντιστοιχιστεί το ερώτημα σε ένα πρότυπο. Το παραπάνω ερώτημα θα αναγνωρίσει τις δυο οντότητες και θα χτίσει το ακόλουθο ερώτημα. Αρχικά αναζήτηση στον index actors για να βρει την εγγραφή με

Name ‘Morgan Freeman’, κάνοντας χρήση fuzzy search για την αντιστοίχιση σε περίπτωση ορθογραφικού λάθους. Αφού εντοπιστεί το id από των πίνακα των ηθοποιών, γίνεται αναζήτηση με το id στον index titles όπου εφαρμόζεται το facet action στο πεδίο genres και με limit 10 ταξινομώντας με το rating κατά φθίνουσα σειρά.

Στην συνέχει παρουσιάζεται στην Εικόνα 20 η σελίδα αποτελεσμάτων για το ερώτημα ‘find {genre} movies from year {number}’. Όπου genre: ‘harror’, number:’1995’. Παρατηρούμε ότι φέρνει αποτελέσματα κάνοντας fuzzy search, παρακάμπτοντας το ορθογραφικό λάθος. Στα αποτελέσματα αναφέρεται ο τίτλος της ταινίας, η χρονολογία πρώτης προβολής, τα λεπτά προβολής, σύνδεσμος για την ιστοσελίδα του imdb που αντιστοιχεί στην ταινία, ώστε να μπορούμε να επιβεβαιώσουμε άμεσα την ορθότητα του αποτελέσματος και τέλος την κατηγορία της ταινίας (πχ Drama, horror …)



Εικόνα 21 Αποτελεσμάτων για ερώτημα 'find {genre} movies from year {number}'

6

Επίλογος

Θα συνεχίσουμε και θα κλείσουμε την εργασία με την σύνοψη και τα συμπεράσματα τα οποία έχουμε εξάγει από την ενδελεχή μελέτη της ανάκτησης πληροφορίας, μέσω μηχανών αναζήτησης στις οποίες γίνεται εκτεταμένη χρήση ταξινομιών και τεχνικών επεξεργασίας φυσικής γλώσσας.

6.1 Σύνοψη και συμπεράσματα

Στόχος της πτυχιακής ήταν η υλοποίηση διαδικτυακής μηχανής αναζήτησης η οποία δέχεται ερωτήματα που δίνονται σε ‘φυσική γλώσσα’ από τον χρήστη, βασιζόμενη σε τεχνικές NLP (Natural Language Processing), στην συνέχεια επεξεργάζεται τα ερωτήματα συντακτικά, σημασιολογικά, μορφολογικά και γραμματικά τα ερωτήματα και ακολούθως προσπαθεί να αντιστοιχήσει το ερώτημα σε ένα συγκεκριμένο facet ανακτώντας με τον τρόπο αυτό μόνο τις σχετικές με το ερώτημα εγγραφές. Για την επίτευξη του στόχου αυτού απαιτήθηκε η μελέτη των γνωστικών πεδίων της επεξεργασία φυσικής γλώσσα και της ανάκτησης πληροφοριών.

Κατόπιν μελέτης σε βάθος του θεωρητικού υπόβαθρου των δύο παραπάνω τομέων, καθώς και έρευνας για τις δυνατότητες των διαθέσιμων εργαλείων για την υλοποίηση της εφαρμογής, καταλήξαμε στην χρήση των Google Natural Language Processing και Elasticsearch. Τα συστήματα αυτά συγκέντρωναν αρκετά πλεονεκτήματα σε σχέση με τον ανταγωνισμό τους. Και τα δυο μας παρέχουν διεπαφή Restfull API, κάνοντας χρήση Json μορφοποίησης των διακινούμενων δεδομένων. Ως αποτέλεσμα αποτελούν ιδανικά εργαλεία που μας επιτρέπουν την πρόσβαση από οποιοδήποτε σέρβερ, με την χρήση οποιασδήποτε προγραμματιστικής γλώσσας, κάτι ιδιαίτερος επιθυμητό στο πεδίο των διαδικτυακών εφαρμογών. Το χαρακτηριστικό αυτό βοήθησε ιδιαιτέρως στην απρόσκοπτη επικοινωνία των δυο βασικών κομματιών της εφαρμογής.

Το Elasticsearch αποτέλεσε την βάση του συστήματος μας, παρέχοντας μια μηχανή αναζήτησης με εξελιγμένες δυνατότητες. Οι δυνατότητες κλιμάκωσης της μηχανής αναζήτησης για χρήση με τεράστιο όγκο δεδομένων, η ταχύτητα αναζήτησης χάρη σε χρήση B-trees αποτελούν μεγάλα πλεονεκτήματα στις διαδικτυακές εφαρμογές. Βασικότερα στοιχεία αποτελούν οι δυνατότητες full text search, faceted search ορθογραφικού ελέγχου και ασαφούς αναζήτησης. Οι λειτουργίες αυτές συνιστούν πλέον προαπαιτούμενα χαρακτηριστικά για τις μηχανές αναζήτησης στο σύγχρονο περιβάλλον.

Η χρήση ταξινομιών για την αναπαράσταση του corpus στις μηχανές αναζήτησης, αποτελεί ένα μέσο βελτιστοποίησης της αναζήτησης. Με τον τρόπο αυτό απομακρυνόμαστε από την κλασική full text αναζήτηση, πηγαίνοντας προς ένα μίγμα κλασικής αναζήτησης πλήρους κειμένου και faceted search, το οποίο έχει ως αποτέλεσμα μεγαλύτερη ακρίβεια, ταχύτητα καθώς και εξελιγμένες δυνατότητες query expansion και query suggestion.

Στο σύγχρονο περιβάλλον είναι πλέον επιβεβλημένη η χρήση τεχνικών επεξεργασίας φυσικής γλώσσας για την βέλτιστη εμπειρία χρήσης των μηχανών αναζήτησης από τους χρήστες. Η επιλογή του Google Natural Language Processing προσέφερε εξελιγμένες δυνατότητες ανάλυσης φυσικής γλώσσας στην εφαρμογή μας. Με τον τρόπο αυτό μπορέσαμε να έχουμε μορφολογική, συντακτική, σημασιολογική και συναισθηματική ανάλυση των ερωτημάτων αναζήτησης. Επιπρόσθετα οι λειτουργίες εξαγωγής οντοτήτων, NER named entity recognition και τα δέντρα εξαρτήσεων αποτέλεσαν πολύτιμα στοιχεία για την επίτευξη του στόχου μας. Με βάση τα παραπάνω στοιχεία μπορέσαμε να καθορίσουμε τα απαραίτητα facet ώστε να εκτελέσουμε τα ερωτήματα και να πετύχουμε τους στόχους της υλοποίησης. Λόγο του μικρού μήκους των ερωτημάτων, δεν μπορέσαμε να πάρουμε δεδομένα σχετικά με την κατηγοριοποίηση κειμένου, αν και στην περίπτωση του συγκεκριμένου corpus δεν θα παρείχαν κάποια επιπρόσθετη πληροφορία.

Βασικό κομμάτι για την αναζήτηση σε μηχανές αναζήτησης με χρήση ταξινομιών είναι η δημιουργία προτύπων με βάση τα παραπάνω στοιχεία, για την αντιστοίχιση του ερωτήματος του χρήστη σε ερώτημα για την μηχανή αναζήτησης, το οποίο κάνει χρήση συνδυασμού facet και ερωτημάτων πλήρους κειμένου. Η επιλογή της συγγραφής τέτοιον προτύπων, αντί για την δημιουργία τους μέσω στατιστικών ή μέσω μελέτης των ερωτημάτων χρηστών αποτέλεσε μονόδρομο καθώς δεν είχαμε στην διάθεση μας τέτοια δεδομένα. Η συγγραφή των προτύπων αποτελεί μια καλή λύση σε συστήματα τα οποία έχουν ένα γνωστικό πεδίο και αποτελεί μια γρήγορη και οικονομικότερη λύση. Επιπρόσθετα σε περιπτώσεις εμπορικών εφαρμογών παρέχει την δυνατότητα εφαρμογής εμπορικών πολιτικών.

Εν κατακλείδι η έρευνα μας ένωσε τα κομμάτια της επεξεργασίας φυσικής γλώσσας και της ανάκτησης πληροφορίας, κάνοντας χρήση ταξινομιών, παρέχοντας ένα οδηγό για την ολοκλήρωση των τεχνολογιών, κάνοντας παράλληλα χρήση συστημάτων που παρέχουν

εξελιγμένων τεχνικών για την ανάπτυξη διαδικτυακών και όχι μόνο, μηχανών αναζήτησης. Οι πρακτικές αυτές μπορούν να εφαρμοστούν σε πολλούς τομείς, επί παραδείγματι στο ηλεκτρονικό εμπόριο, σε συστήματα αγγελιών, ευρέσεως εργασίας, καταλόγους ακινήτων, τουριστικών επιχειρήσεων σχεδόν στο σύνολο των περιπτώσεων αναζήτησης.

6.2 Μελλοντικές επεκτάσεις

Η έρευνα και η υλοποίηση εστιάστηκε στην μελέτη μηχανών αναζήτησης με τεχνικές επεξεργασίας φυσικής γλώσσας και με χρήση ταξινομιών. Κατά την έρευνα εντοπίσαμε ότι η χρήση οντολογιών αντί ταξινομιών θα μπορούσε να μας παρέχει επιπρόσθετα πλεονεκτήματα καθώς και να βελτιώσει την ακρίβεια των μηχανών αναζήτησης.

Η διεύρυνση της χρήσης σημασιολογικών δεδομένων στον σύνολο της πληροφορικής, είναι δεδομένη. Το όραμα του σημασιολογικού ιστού προχωράει με σταθερά βήματα προς την ολοκλήρωση, αν και απέχουμε από το όραμα του Tim Berners Lee, είναι πλέον κοινώς αποδεκτό ότι τα σημασιολογικά δεδομένα και κατ' επέκταση οι οντολογίες μπορούν να μας παρέχουν επιπρόσθετη πληροφορία.

Στην περίπτωση της εφαρμογής που μελετούμε θα μπορούσαν να βοηθήσουν σε σημαντικό βαθμό. Αρχικά με την χρήση οντολογιών συγκεκριμένου τομέα, όπως είναι η περίπτωση η οποία μελετήθηκε στην έρευνα μας, θα μπορούσαμε να ορίσουμε συνώνυμα, και μέσω των τριπλετών να ορίσουμε σχέσεις ανάμεσα στις διάφορες οντότητες του πεδίου εφαρμογής. Με τον τρόπο αυτό θα μπορούσαμε να παράγουμε ευκολότερα τα πρότυπα αντιστοίχισης των ερωτημάτων των χρηστών σε ερωτήματα στην μηχανή αναζήτησης ή ακόμη και να παράγουμε αυτόματα τα πρότυπα από την οντολογία. Θα μπορούσαμε ακόμη να αποκομίσουμε κέρδη στο κομμάτι της παραγωγής συστάσεων και query expansion. Επιπρόσθετα θα μπορούσαμε να παράγουμε αυτοματοποιημένα τα facet του corpus της εφαρμογής μας.

Τα συστήματα διαλόγου αποτελούν ένα ακόμη τομέα ο οποίος μπορεί να αποτελέσει αντικείμενο επέκτασης της εφαρμογή μας. Τα συστήματα αυτά εφάπτονται του τομέα της επεξεργασίας φυσικής γλώσσας. Παρέχουν καθοδήγηση στον χρήστη, ώστε να μπορέσει να ανακτήσει την πληροφορία που αναζητά. Μια τέτοια απαίτηση γίνεται ολοένα και πιο επιτακτική από τα σύγχρονα συστήματα καθώς βοηθάν ιδιαίτερα τον χρήστη που δεν έχει ιδιαίτερο τεχνολογικό υπόβαθρο ή γνώση του πεδίου εφαρμογής ενώ συστήματος, να έχει καλύτερη εμπειρία χρήσης των συστημάτων. Τα συστήματα διαλόγου θα μπορούσαν να εμπλουτιστούν και αυτά με σημασιολογικά δεδομένα ώστε να επιτύχουν καλύτερα αποτελέσματα.

7

Παράρτημα

7.1 Κώδικας mapping index ηθοποιών

```
-XPOST http://195.201.112.157:9200/actors/ -d '{
            "mappings":{
        "persons":{
          "_all":{
            "analyzer":"english_nGram_analyzer",
            "search_analyzer":"english_whitespace_analyzer"
          },
          "properties":{
            "actor_id":{
              "type":"long"
            },
            "name":{
              "type":"string",
              "index":"not_analyzed"
            },
            "birthyear":{
              "type":"long"
            },
            "deathyear":{
              "type":"long"
            },
            "primaryProfession":{
```

```
            "type":"string",
            "index":"not_analyzed"
          },
          "knownForTitles":{
            "type":"string",
            "index":"not_analyzed"
          }
        }
}'
```

7.2 Κώδικας settings index ηθοποιών

```
-XPOST http://195.201.112.157:9200/actors/ -d '{
    "settings":{
      "index":{
        "analysis":{
          "filter":{
            "english_stop":{
              "type":"stop",
              "stopwords":"_english_"
            },
            "english_lowercase":{
              "type":"lowercase"
            },
            "nGram_filter":{
              "token_chars":[
                "letter",
                "digit",
                "punctuation",
                "symbol"
              ],
              "min_gram":"2",
              "type":"nGram",
              "max_gram":"20"
```

```
    }
  },
  "analyzer":{
    "nGram_analyzer":{
      "filter":[
        "lowercase",
        "asciifolding",
        "nGram_filter"
      ],
      "type":"custom",
      "tokenizer":"whitespace"
    },
    "whitespace_analyzer":{
      "filter":[
        "asciifolding",
        "lowercase"
      ],
      "type":"custom",
      "tokenizer":"whitespace"
    },
    "english_nGram_analyzer":{
      "filter":[
        "english_lowercase",
        "english_stop",
        "nGram_filter"
      ],
      "type":"custom",
      "tokenizer":"whitespace"
    },
    "english_whitespace_analyzer":{
      "filter":[
        "lowercase"
      ],
      "type":"custom",
```

```
            "tokenizer":"whitespace"
          }
        },
        "char_filter":{
          "replace":{
            "type":"mapping",
            "mappings":[
              "& => and",
              "% => percent",
              "$ => dollar"
            ]
          }
        }
      },
      "number_of_shards":"5",
      "number_of_replicas":"1"
    }
  }
```

7.3 Κώδικας mapping index τίτλων

```
-XPOST http://195.201.112.157:9200/titles/ -d '{
          "mappings":{
      "show":{
        "_all":{
          "analyzer":"english_nGram_analyzer",
          "search_analyzer":"english_whitespace_analyzer"
        },
        "properties":{
          "title_id":{
            "type":"long"
          },
          "titletype":{
            "type":"string",
            "index":"not_analyzed"
```

```
        },
        "primarytitle":{
          "type":"string",
          "index":"not_analyzed"
        },
        "originaltitle":{
          "type":"string",
          "index":"not_analyzed"
        },
                        "isadult":{
          "type":"long"
        },
                        "startyear":{
          "type":"long"
        },
                        "endyear":{
          "type":"long"
        },
                        "runtimeminutes":{
          "type":"long"
        },
        "genres":{
          "type":"string",
          "index":"not_analyzed"
        }
      }
    }
  }
}'
```

7.4 Κώδικας settings index τίτλων

```
 -XPOST http://195.201.112.157:9200/titles/ -d '{
    "settings":{
      "index":{
```

```
"analysis":{
  "filter":{
    "english_stop":{
      "type":"stop",
      "stopwords":"_english_"
    },
    "english_lowercase":{
      "type":"lowercase"
    },
    "nGram_filter":{
      "token_chars":[
        "letter",
        "digit",
        "punctuation",
        "symbol"
      ],
      "min_gram":"2",
      "type":"nGram",
      "max_gram":"20"
    }
  },
  "analyzer":{
    "nGram_analyzer":{
      "filter":[
        "lowercase",
        "asciifolding",
        "nGram_filter"
      ],
      "type":"custom",
      "tokenizer":"whitespace"
    },
    "whitespace_analyzer":{
      "filter":[
        "asciifolding",
```

```
            "lowercase"
          ],
          "type":"custom",
          "tokenizer":"whitespace"
        },
        "english_nGram_analyzer":{
          "filter":[
            "english_lowercase",
            "english_stop",
            "nGram_filter"
          ],
          "type":"custom",
          "tokenizer":"whitespace"
        },
        "english_whitespace_analyzer":{
          "filter":[
            "lowercase"
          ],
          "type":"custom",
          "tokenizer":"whitespace"
        }
      },
      "char_filter":{
        "replace":{
          "type":"mapping",
          "mappings":[
            "& => and",
            "% => percent",
            "$ => dollar"
          ]
        }
      }
    },
    "number_of_shards":"5",
```

```
        "number_of_replicas":"1"
      }
    },
}'
```

8

Βιβλιογραφία

[1] Chomsky, Noam , Aspects of the Theory of Syntax, Cambridge, Massachusetts: MIT Press , 1965

[2] Rospocher, M., van Erp, M., Vossen, P., Fokkens, A., Aldabe,I., Rigau, G., Soroa, A., Ploeger, T., and Bogaard, T.(2016). Building event-centric knowledge graphs from news. Web Semantics: Science, Services and Agents on the World Wide Web, In Press.

[3] Shemtov, H. (1997). Ambiguity management in natural language generation. Stanford University.

[4] Emele, M. C., & Dorna, M. (1998, August). Ambiguity preserving machine translation using packed representations. In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1 (pp. 365-371). Association for Computational Linguistics.

[5] Knight, K., & Langkilde, I. (2000, July). Preserving ambiguities in generation via automata intersection. In AAAI/IAAI (pp. 697-702).

[6] Nation, K., Snowling, M. J., & Clarke, P. (2007). Dissecting the relationship between language skills and learning to read: Semantic and phonological contributions to new vocabulary learning in children with poor reading comprehension. Advances in Speech Language Pathology, 9(2), 131-139.

[7] Elizabeth D. Liddy (2001). Natural language processing.

[8] Stuart J. Russell and Peter Norvig (2003), 'Artificial Intelligence: A Modern Approach'

[9] Kamp, H., & Reyle, U. (1993). Tense and Aspect. In From Discourse to Logic (pp. 483-689). Springer Netherlands.

[10] Hayes, P. J. (1992). Intelligent high-volume text processing using shallow, domain-specific techniques. Text-based intelligent systems: Current research and practice in information extraction and retrieval, 227-242.

[11] Lewis, D. D. (1998, April). Naive (Bayes) at forty: The independence assumption in information retrieval. In European conference on machine learning (pp. 4-15). Springer Berlin Heidelberg

[12] Nitin Indurkhya Fred J. Damerau (2010), Handbook of Natural Language Processing, Chapman & Hall/CRC

[13] Emdad Khan (2016), Machine Learning Algorithms for Natural Language Semantics and Cognitive Computing, 2016 International Conference on Computational Science and Computational Intelligence (CSCI)

[14] Tillmann, C., Vogel, S., Ney, H., Zubiaga, A., & Sawaf, H. (1997, September). Accelerated DP based search for statistical translation. In Eurospeech.

[15] Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002, July). BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association fo computational linguistics (pp. 311-318). Association for Computational Linguistics

[16] Sakkis, G., Androutsopoulos, I., Paliouras, G., Karkaletsis, V., Spyropoulos, C. D., & Stamatopoulos, P. (2001). Stacking classifiers for anti-spam filtering of e-mail.

[17] McCallum, A., & Nigam, K. (1998, July). A comparison of event models for naive bayes text classification. In AAAI-98 workshop on learning for text categorization (Vol. 752, pp. 41-48).

[18] Hayes, P. J. (1992). Intelligent high-volume text processing using shallow, domain-specific techniques. Text-based intelligent systems: Current research and practice in information extraction and retrieval, 227-242

[19] Zajic, D. M., Dorr, B. J., & Lin, J. (2008). Single-document and multi-document summarization techniques for email threads using sentence compression. Information Processing & Management, 44(4), 1600-1610.

[20] Fattah, M. A., & Ren, F. (2009). GA, MR, FFNN, PNN and GMM based models for automatic text summarization. Computer Speech & Language, 23(1), 126-144.

[21] Goffman, W. (1964). On relevance as a measure. Information Storage and Retrieval 2: pp. 201–203.

[22] Mizzaro, S. (1997) (September). Relevance: the whole history. Journal of the American Society for Information Science

[23] Saracevic, T. (2006). Relevance: a review of the literature and a framework for thinking on the notion in information science. Part II. Advances in Librarianship

[24] https://trec.nist.gov/

[25] Manning, C., Raghavan, P., and Schütze, H. (2008). Introduction to Information Retrieval. New York: Cambridge University Press

[26] Spink, A., Wolfram, D., Jansen, M. B., and Saracevic, T. (2001). Searching the Web: the public and their queries. Journal of the American Society for Information Science and Technology

[27] Robertson, S. (2000) (April). Salton Award Lecture on theoretical argument in information retrieval. SIGIR Forum

[28] Turtle, H. (1994). Natural language vs. Boolean query evaluation: a comparison of retrieval performance. In Proceedings of the 17th Annual international ACM SIGIR Conference on Research and Development in information Retrieval

[29] Makhoul, J., Kubala, F., Schwartz, R., Weischedel (1999) R.: Performance measures for information extraction. In: Proceedings of DARPA Broadcast News Workshop, Herndon

[30] Mooers, C. (1950). Coding, information retrieval, and the rapid selector. American Documentation

[31] Mooers, C. (1961). From a point of view of mathematical etc. techniques. In R. A. Fairthorne (Ed.), Towards Information Retrieval

[32] Salton, G., Wong, A., and Yang, C. S. (1975), A Vector Space Model for Automatic Indexing. Communications of the ACM

[33] Spärck Jones, K. 1972. A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation

[34] Deerwester, S., Dumais, S., Furnas, G. W., Landauer, T. K., and Harshman, R. 1990. Indexing by latent semantic analysis. Journal of the American Society for Information Science

[35] Kleinberg, J. 1999. Authoritative sources in a hyperlinked environment. Journal of the ACM

[36] Brin, S., and Page, L. 1998. The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems

[37] Dewey Decimal Classification and Relative Index. Dublin, OH: OCLC Forest Press

[38] Gruber, T. (1993). A translation approach to portable ontology specifications. Knowledge Acquisition

[39] Smith, M., Welty, C., and McGuinness, D. L. 2004. OWL Web Ontology Language Guide. W3C

[40] Guarino, N. (1998) Formal Ontology in Information Systems: Proceedings of the First International Conference (Fois'98), June 6-8, Trento, Italy, Ios Pr Inc

[41] Herman, I. 2008. Semantic Web Activity Statement. W3C

[42] M., F. Orciuoli, S. Paolozzi and S. Salerno (2011) Ontology Extraction for Knowledge Reuse: The E-Learning Perspective. Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE

[43] Li, Z. and K. Ramani (2007) Ontology-Based Design Information Extraction and Retrieval. AI EDAM

[44] https://cloud.google.com/natural-language/

[45] https://cloud.google.com/docs/

[46] https://ai.google/

[47] https://datasets.imdbws.com/

[48] https://www.imdb.com/interfaces/

[49] https://www.elastic.co/

[50] http://lucene.apache.org/

[51] Edwood Ng, Vineeth Mohan (2015), Lucene 4 Cookbook, Packt Publishing

[52] https://www.json.org/

[53] Clinton Gormley Zachary Tong (2015), Elasticsearch: The Definitive Guide: A Distributed Real-Time Search and Analytics Engine, O'Reilly Media

[54] Rafal Kuc, Marek Rogozinski (2016), Elasticsearch Server, Packt Publishing

[55] https://www.elastic.co/guide/en/elasticsearch/client/php-api/2.0/_quickstart.html

[56] https://www.oracle.com/technetwork/java/javase/overview/java8-2100321.html

[57] http://php.net/manual/en/migration70.new-features.php

[58] https://httpd.apache.org/

[59] https://www.elastic.co/guide/en/elasticsearch/reference/2.4/query-dsl-fuzzy-query.html