Π3.2 Αναφορά Τεκμηρίωσης

DATE: 30/06/2020

ΔΡΑΣΗ ΕΘΝΙΚΗΣ ΕΜΒΕΛΕΙΑΣ «ΕΡΕΥΝΩ-ΔΗΜΙΟΥΡΓΩ-ΚΑΙΝΟΤΟΜΩ»

Τ1ΕΔΚ-05094 - Λεξιπαίγνιο

Ανάπτυξη προηγμένου, καινοτόμου και state-of-the-art υπολογιστικού περιβάλλοντος, για τη δημιουργία ψηφιακών εκπαιδευτικών παιχνιδιών που απευθύνονται σε μαθητές με στόχο: α) τη βελτίωση του γλωσσικού και γενικότερα μαθησιακού τους επιπέδου και β) την ανάπτυξη ποικίλων δεξιοτήτων τους σχετικά με το λεξιλόγιο και τη γλώσσα γενικότερα, αλλά και το περιεχόμενο ειδικών γνωστικών αντικειμένων (π.χ. γεωλογία-γεωγραφία, βιολογία κ.ά.).

NEUROCOM

ITYE "ΔΙΟΦΑΝΤΟΣ" ινστιτούτο τεχνολογίας υπολογιστών & εκδόσεων

ΗΜΕΡΟΜΗΝΙΑ ΕΝΑΡΞΗΣ ΕΡΓΟΥ	09/07/2018
ΔΙΑΡΚΕΙΑ ΕΡΓΟΥ	40 Μήνες

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Ταμείο Περιφερειακής Ανάπτυξης

ΕΠΑνΕΚ 2014-2020
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΑΝΤΑΓΩΝΙΣΤΙΚΟΤΗΤΑ
ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΤΗΤΑ
ΚΑΙΝΟΤΟΜΙΑ

ΕΣΠΑ 2014-2020
ανάπτυξη - εργασία - αλληλεγγύη

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Π3.2 Αναφορά Τεκμηρίωσης	
ΑΡΧΙΚΗ ΗΜΕΡΟΜΗΝΙΑ ΠΑΡΑΔΟΣΗΣ:	05/2020
ΠΡΑΓΜΑΤΙΚΗ ΗΜΕΡΟΜΗΝΙΑ ΠΑΡΑΔΟΣΗΣ:	06/2020
ΕΙΔΟΣ ΠΑΡΑΔΟΤΕΟΥ:	Έκθεση
ΣΥΝΤΑΚΤΗΣ:	Δρ. Αρ. Βαγγελάτος (ΙΤΥΕ)
ΕΚΔΟΣΗ ΕΓΓΡΑΦΟΥ:	v. 1.21

ΣΥΝΤΕΛΕΣΤΕΣ ΠΑΡΑΔΟΤΕΟΥ	
Όνομα	**Φορέας**
Αριστείδης Βαγγελάτος	ΙΤΥΕ
Μόνικα Γαβριηλίδου	ΙΤΥΕ
Μαρία Φουντάνα	ΙΤΥΕ
Άννα Ιορδανίδου	ΙΤΥΕ
Νίκος Αλεξόπουλος	ΙΤΥΕ
Χρήστος Τσαλίδης	NEUROCOM
Βασίλης Σπιταδάκης	NEUROCOM
Ιωάννης Σταματόπουλος	NEUROCOM
Γιώργος Παναγιωτάκης	NEUROCOM
Μαρία Φωστιέρη	NEUROCOM
Γιάννης Καφούσης	NEUROCOM
Αναστάσιος Καλαενζτής	NEUROCOM
Κωνσταντίνος Αλεξάκης	NEUROCOM

«Το παραδοτέο υλοποιήθηκε στο πλαίσιο της Δράσης ΕΡΕΥΝΩ – ΔΗΜΙΟΥΡΓΩ - ΚΑΙΝΟΤΟΜΩ και συγχρηματοδοτήθηκε από την Ευρωπαϊκή Ένωση και εθνικούς πόρους μέσω του Ε.Π. Ανταγωνιστικότητα, Επιχειρηματικότητα & Καινοτομία (ΕΠΑνΕΚ) (κωδικός έργου:Τ1ΕΔΚ-05194)»

Πίνακας Περιεχομένων

Εισαγωγή

Το έργο ΛΕΞΙΠΑΙΓΝΙΟ, είναι ένα ερευνητικό έργο, ενταγμένο στη Δράση ΕΡΕΥΝΩ – ΔΗΜΙΟΥΡΓΩ - ΚΑΙΝΟΤΟΜΩ που συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση και εθνικούς πόρους μέσω του Ε.Π. «Ανταγωνιστικότητα, Επιχειρηματικότητα & Καινοτομία» (ΕΠΑνΕΚ) (κωδικός έργου:Τ1ΕΔΚ-05194). Για την υλοποίησή του συνεργάζονται δύο φορείς: το ερευνητικό Ινστιτούτο Τεχνολογίας Υπολογιστών και Εκδόσεων (ΙΤΥΕ) και η εταιρεία Neurocom Α.Ε.

Στόχος του έργου είναι η Ανάπτυξη προηγμένου, καινοτόμου και state-of-the-art υπολογιστικού περιβάλλοντος, για τη δημιουργία ψηφιακών εκπαιδευτικών παιχνιδιών που απευθύνονται σε μαθητές με σκοπό: α) τη βελτίωση του γλωσσικού και γενικότερα μαθησιακού τους επιπέδου και β) την ανάπτυξη ποικίλων δεξιοτήτων τους σχετικά με το λεξιλόγιο και τη γλώσσα γενικότερα, αλλά και το περιεχόμενο ειδικών γνωστικών αντικειμένων (π.χ. γεωλογία-γεωγραφία, βιολογία, κ.α.).

Το παρόν παραδοτέο, Π3.2: *Αναφορά Τεκμηρίωσης*, περιγράφει και τεκμηριώνει την υλοποίηση της Ενότητας Εργασίας 3: «Επέκταση – χαρακτηρισμός του γλωσσικού υλικού των παιχνιδιών», όπως και το ίδιο το γλωσσικό υλικό που περιλαμβάνεται στο Παραδοτέο Π3.3: *Χαρακτηρισμένο Γλωσσικό Υλικό*.

Με βάση τόσο τον «τεχνικό σχεδιασμό» του έργου, όσο και τις πραγματικές ανάγκες του, στο πλαίσιο της ΕΕ3, έγινε αρχικά μια αποτίμηση των προϋπαρχόντων πόρων και στη συνέχεια υλοποιήθηκαν οι αναγκαίες βελτιώσεις/προσθήκες. Κατόπιν έγινε ταξινόμηση και χαρακτηρισμός του υλικού που περιλαμβάνεται στους υπάρχοντες αλλά και τους νέους γλωσσικούς πόρους με βάση τους καταλόγους των γραμματικών φαινομένων και γλωσσικών λαθών όπως αυτοί προέκυψαν από την ΕΕ1.

Το αποτέλεσμα ήταν μια σύνθετη και εκτενής βάση γλωσσικών δεδομένων, από την οποία μπορεί να προκύπτει δυναμικά είσοδος για τα παιχνίδια που σχεδιάζονται/υλοποιούνται με τρόπο παραμετρικό, με παραμέτρους που θέτει είτε το ίδιο το παιχνίδι (με βάση τα χαρακτηριστικά και τις απαιτήσεις του), είτε ο χρήστης (εκπαιδευτικός/μαθητής).

Στα επόμενα, περιγράφονται με τη σειρά οι προϋπάρχοντες πόροι, οι νέοι/εμπλουτισμένοι πόροι, οι πηγές και τα εργαλεία που χρησιμοποιήθηκαν αλλά και τα περιεχόμενα του παραδοτέου Π3.3 (το χαρακτηρισμένο γλωσσικό υλικό). Στο τέλος παρατίθενται τα παραρτήματα. Τμήματα των παραρτημάτων περιλαμβάνονται και στο Π3.2 για λόγους συνέπειας.

1. Προϋπάρχοντες πόροι

Στο πλαίσιο του έργου, όπως περιγράφηκε και στην αρχική πρόταση / Τεχνικό Δελτίο, στόχος ήταν η αξιοποίηση υπαρχόντων γλωσσικών πόρων/υποδομών, με παράλληλη εξέλιξη και βελτίωσή τους: «*Η εταιρεία Neurocom Α.Ε. αλλά και το ΙΤΥΕ «Διόφαντος» έχουν αναπτύξει και διαθέτουν σημαντικές υποδομές που αφορούν σε πόρους και εργαλεία επεξεργασίας φυσικής γλώσσας, όπως και συσσωρευτές εκπαιδευτικού περιεχομένου και εκπαιδευτικές υποδομές. Μέχρι σήμερα έχουν υλοποιηθεί μία σειρά εργαλείων γλωσσικού ελέγχου (ορθογράφος, συλλαβιστής, θησαυρός συνωνύμων, γραμματικός διορθωτής) αλλά και σύνθετων γλωσσικών εργαλείων με ειδικό εκπαιδευτικό βάρος, που αναβαθμίζουν τη γλωσσική διδασκαλία με τη συνδρομή των ΤΠΕ. Η υλοποίηση του έργου θα στηριχθεί σε εξέλιξη, βελτίωση αξιοποίηση και περαιτέρω ανάπτυξη των παραπάνω υποδομών*».

Στο πλαίσιο αυτό καταγράφονται στη συνέχεια οι προϋπάρχοντες πόροι/υποδομές που αξιοποιούνται στο έργο.

Μορφολογικό Λεξικό Ελληνικής γλώσσας:

Το λεξικό αυτό είναι αποτέλεσμα μιας μακράς ανάπτυξης τόσο σε χρόνο όσο και σε ανθρωπο-προσπάθεια (Vagelatos et al, 1995, Ntoulas et al., 2000, Tsalidis et al, 2004, Gakis et al. 2012) αλλά και βάση για προσπάθειες σε άλλους τομείς (π.χ. στο βιοιατρικό τομέα – Tsalidis et al., 2003, Vagelatos et al., 2007). Αποτελείται από περίπου 90.000 λήμματα, με πληροφορία:

- *Ορθογραφική*: ποια γράμματα και με ποια σειρά αποτελούν την ορθή γραφή του κλιτικού τύπου
- *Συλλαβισμού*: ποιες συλλαβές απαρτίζουν τον κλιτικό τύπο (έχει εξελιχθεί και σχετικό αυτόνομο εργαλείο: Συλλαβιστής)
- *Μορφηματική*: ποια μορφήματα (πρόθημα, θέμα, επίθημα, κατάληξη) απαρτίζουν τον κλιτικό τύπο
- *Μορφοσυντακτική*: από ποια λέξη προέρχεται ο κλιτικός τύπος και ποια είναι τα μορφοσυντακτικά χαρακτηριστικά του (μέρος του λόγου, γένος, πτώση, αριθμός, πρόσωπο, φωνή, χρόνος κτλ.)
- *Υφολογική*: αν υπάρχουν, ποια είναι τα υφολογικά χαρακτηριστικά του κλιτικού τύπου, π.χ. ο τύπος *δίνανε* χαρακτηρίζεται υφολογικά ως «προφορικός» (σε αντιδιαστολή προς τον τύπο *έδιναν*)
- *Ορολογική*: αν ο κλιτικός τύπος αποτελεί ειδικό όρο, σε ποιο ειδικό λεξιλόγιο ανήκει, π.χ. ο τύπος *αβιογένεση* αποτελεί όρο της Βιολογίας.

Πέραν των κοινών λέξεων, το Μορφολογικό Λεξικό περιλαμβάνει και λέξεις ειδικού λεξιλογίου, όπως 10.000 ελληνικά τοπωνύμια (ονόματα ελληνικών νομών, δήμων, κοινοτήτων, επαρχιών, πόλεων, χωριών κτλ.), κ.α.

Θησαυρός Συνωνύμων – Αντιθέτων

Ο Θησαυρός είναι ένα ειδικού τύπου λεξικό το οποίο επιχειρεί να αποδώσει τις σημασίες λέξεων ή εκφράσεων της νέας ελληνικής μέσω συνωνύμων, αντιθέτων και παραδειγμάτων χρήσης (Tsalidis et al., 2004). Τα κύρια χαρακτηριστικά του Θησαυρού είναι:

- Περιέχει ~22.000 λήμματα: 11.500 ουσιαστικά, 5.150 επίθετα, 3.500 ρήματα, 650 εκφράσεις και 1.200 επιρρήματα, προθέσεις και συνδέσμους.

- Σε κάθε λήμμα υπάρχει διάκριση σημασιών. Τα συνώνυμα, τα αντίθετα και τα παραδείγματα χρήσης ομαδοποιούνται ανά σημασία.

- Οι λημματικοί τύποι, οι σημασίες τους και τα συνώνυμα/αντίθετα ανά σημασία συνοδεύονται από υφολογικές και πραγματολογικές πληροφορίες.
- Για οποιαδήποτε λέξη ή φράση, η οποία παίζει το ρόλο συνωνύμου, αντιθέτου ή σχετικής έκφρασης μέσα σε ένα λήμμα, υπάρχει σχεδόν πάντοτε (96%) λήμμα του Θησαυρού που την περιγράφει.
- Κάθε αναφορά σε πολύσημο συνώνυμο ή αντίθετο συνοδεύεται από αριθμητικό δείκτη, ο οποίος δηλώνει τον αριθμό της σημασίας μέσα στο λήμμα του συνωνύμου ή αντιθέτου.
- Τα λήμματα συνοδεύονται από αναφορές σε σχετικές εκφράσεις (αν υπάρχουν).

Χτίζω Λέξεις

Το πρωτότυπο αυτό λεξικό (Ιορδανίδου & Πανταζάρα, 2010) παρουσιάζει τους τρόπους με τους οποίους «χτίζονται» οι νεοελληνικές λέξεις, ποια είναι δηλαδή τα συστατικά τους και πώς συνδυάζονται μεταξύ τους. Περιέχει αναλυτικά σχολιασμένες σημασίες, παραδείγματα χρήσης, συνώνυμα-αντίθετα και σημειώσεις για την ορθογραφία και την ετυμολογία. Αποτελεί σημαντική πηγή πληροφόρησης για το λεξιλόγιο της νέας ελληνικής, και ειδικότερα μπορεί να χρησιμεύσει στο σχεδιασμό και στη λύση γλωσσικών ασκήσεων. Όπως επιβάλλει η σύγχρονη λεξικογραφία, η ανάπτυξη του λεξικού έγινε ηλεκτρονικά με τη βοήθεια λεξικογραφικής βάσης δεδομένων της Neurolingo.

Στόχος του λεξικού είναι να διευκολύνει το μαθητή, το σπουδαστή (και γενικότερα το χρήστη της νέας ελληνικής) να κατανοήσει τον τρόπο παραγωγής και σύνθεσης των νεοελληνικών λέξεων. Για να αποφύγουμε ζητήματα γλωσσολογικής υφής και ορολογίας, υιοθετήσαμε τον όρο α΄ συστατικά (για προθήματα, συμφύματα ή προσφυματοειδή, α΄ συνθετικά) και β΄ συστατικά (για επιθήματα, δεσμευμένα θέματα, β΄ συνθετικά).

Λεξισκόπιο

«Μάθετε για την ορθογραφία, τη μορφολογία, τον συλλαβισμό και τα συνώνυμα/αντίθετα οποιασδήποτε νεοελληνικής λέξης. Οι λέξεις παραμένουν μάταιοι ήχοι αν δεν μπορούμε να τις κατανοήσουμε.»

Το Λεξισκόπιο (http://www.neurolingo.gr/el/online_tools/lexiscope.htm), είναι μια διαδικτυακή σελίδα μέσω της οποίας, παρουσιάζονται πληροφορίες σχετικά με τον συλλαβισμό, τη μορφολογία (κλίση) και τα συνώνυμα και αντίθετα των λέξεων, αλλά και των πιο γνωστών φράσεων και εκφράσεων στις οποίες μια ελληνική λέξη μπορεί να μετέχει.

Φωτόδεντρο

Το φωτόδεντρο (http://photodentro.edu.gr/aggregator/) ως *εθνικός συσσωρευτής εκπαιδευτικού περιεχομένου* έχει πληθώρα από σημαντικά εκπαιδευτικά αντικείμενα. Πιο συγκεκριμένα, ιδιαίτερο ενδιαφέρον για το έργο παρουσιάζουν:

Α) τα σχολικά βιβλία και ιδιαίτερα τα εμπλουτισμένα, τα οποία περιέχουν επιπλέον στοιχεία κατά περίπτωση (Διαδραστικά σχολικά βιβλία)

Β) Το υλικό (εκπαιδευτικό περιεχόμενο), ειδικά στην κατηγοριοποίησή του ανά θεματική περιοχή (π.χ. Γεωγραφία – γεωλογία, ή Νέα Ελληνική γλώσσα)

Εφαρμογές

Με βάση τους παραπάνω πόρους έχουν αναπτυχθεί μια σειρά εργαλείων επεξεργασίας φυσικής γλώσσας, τα βασικότερα τω οποίων είναι τα παρακάτω:

Η πρώτη χρονικά αλλά και σε «ειδικό βάρος» εφαρμογή είναι ο Ορθογράφος (Speller) (Vagelatos et al, 1995). Μέσω αυτού ελέγχονται για προβλήματα ορθογραφίας, λέξεις που πληκτρολογούνται (ή γενικότερα περιέχονται σε ένα π.χ. αρχείο), και εφόσον δεν υπάρχουν στο λεξικό, προτείνονται εναλλακτικές. Για την παραγωγή εναλλακτικών προτάσεων, αξιοποιούνται αλγόριθμοι που βασίζονται στην κατηγοριοποίηση των λαθών (π.χ. τυπογραφικά, ορθογραφικά) αλλά και στα χαρακτηριστικά της Ελληνικής γλώσσας (π.χ. πλούσια μορφολογία, κτλ.).

Μια δεύτερη εφαρμογή είναι ο Λημματοποιητής (Lemmatizer): είναι το γλωσσικό εργαλείο που δέχεται ως είσοδο έναν οποιοδήποτε (ορθό) λεκτικό τύπο και επιστρέφει το λημματικό τύπο στον οποίο αντιστοιχεί, π.χ. για τον τύπο *κατέστη* επιστρέφει το λημματικό τύπο *καθιστώ*. Στην περίπτωση που ο δοθείς τύπος αντιστοιχεί σε περισσότερους του ενός λημματικούς τύπους, ο Λημματοποιητής τους επιστρέφει όλους, π.χ. για τον τύπο *απαντήσεις* επιστρέφει *απαντώ* και *απάντηση*. Για μια γλώσσα με έντονη μορφολογία όπως η Νέα Ελληνική, ο Λημματοποιητής αποτελεί αναπόσπαστη λειτουργική συνιστώσα σε εφαρμογές ευρετηριασμού και αναζήτησης κειμενικής πληροφορίας. Για παράδειγμα, όταν ο χρήστης αναζητά κείμενα που περιέχουν τον όρο *υπολογιστές*, προφανώς θα ήθελε μέσα στα αποτελέσματα αναζήτησης να περιλαμβάνονται και κείμενα που περιέχουν τους τύπους *υπολογιστής*, *υπολογιστή* και *υπολογιστών*. Αυτό πρακτικά επιτυγχάνεται μέσω λημματοποίησης τόσο των όρων ευρετηριασμού των κειμένων όσο και των όρων που απαρτίζουν τα ερωτήματα αναζήτησης: κείμενα που περιέχουν τους τύπους *υπολογιστή*, *υπολογιστές* και *υπολογιστών* θα ευρετηριαστούν και βάσει του λημματικού τύπου *υπολογιστής*· ένα ερώτημα που περιέχει τον τύπο *υπολογιστές* θα διευρυνθεί ώστε να περιέχει και το λημματικό τύπο *υπολογιστής*. Έτσι, ένα κείμενο που περιέχει τον τύπο *υπολογιστών* μπορεί πλέον να συσχετιστεί με ένα ερώτημα που περιέχει τον τύπο *υπολογιστές* μέσω του κοινού λημματικού τύπου *υπολογιστής*.

Μια Τρίτη εφαρμογή είναι ο Συλλαβιστής (Hyphenator): προκειμένου να επιτύχουν πλήρη στοίχιση στις γραμμές μιας παραγράφου, τα συστήματα ηλεκτρονικής στοιχειοθεσίας μεταφέρουν στην επόμενη γραμμή κάθε λέξη που τείνει να βγει εκτός περιθωρίου, ενώ παράλληλα αυξάνουν ή ελαττώνουν ομοιόμορφα τα κενά ανάμεσα στις λέξεις κάθε γραμμής. Η διαδικασία αυτή δεν είναι δυνατό να εφαρμοστεί όταν οι γραμμές έχουν μικρό μήκος (π.χ. γραμμές σε στήλες εφημερίδων) ή όταν στο τέλος των γραμμών βρίσκονται μεγάλες λέξεις, διότι τότε δημιουργούνται μεγάλα κενά σε κάποιες γραμμές, τα οποία επηρεάζουν αρνητικά την αισθητική του κειμένου και αφήνουν πολύ εκτυπώσιμο χώρο ανεκμετάλλευτο. Στην περίπτωση αυτή, η παραδοσιακή πρακτική του συλλαβισμού των λέξεων στο τέλος των γραμμών είναι αναντικατάστατη. Ο ρόλος του *Συλλαβιστή* είναι να υποδεικνύει όλα τα πιθανά σημεία συλλαβισμού μιας λέξης.

Τέλος, υπάρχει ο Γραμματικός Διορθωτής (Grammar Checker). Είναι ένα ηλεκτρονικό εργαλείο σχολιασμού των τμημάτων του κειμένου όπου ενδέχεται:

- να υπάρχει απόκλιση από κανόνα της Σχολικής Γραμματικής, π.χ. στη χρήση του τελικού -ν στον γραπτό λόγο ή στον τονισμό (*ο δάσκαλος μας* αντί του ορθού *ο δάσκαλός μας*)
- να χρησιμοποιείται λανθασμένη γραμματική μορφή λέξης, π.χ. ο τύπος του επιθέτου να μη συμφωνεί με το ουσιαστικό που προσδιορίζει (π.χ. το *αγχώδη* άτομο αντί το *αγχώδες*)
- να εμφανίζεται λανθασμένη χρήση μιας λέξης λόγω σύγχυσης με άλλη (π.χ. *πραγματεύομαι – διαπραγματεύομαι*)
- να υπάρχει λανθασμένη χρήση λέξης στο εσωτερικό μιας φράσης (π.χ. *απορώ και εξανίσταμαι* αντί του ορθού *εξίσταμαι*)

- να αποκλίνει μια συντακτική δομή (π.χ. αποποιήθηκε *της περιουσίας* αντί *την περιουσία*)
- να μην ταιριάζει ένας γλωσσικός τύπος με το ύφος του κειμένου (π.χ. *θεωρούσανε* σε επίσημο ύφος αντί για *θεωρούσαν*)

Ο σχολιασμός που επιστρέφει ο γραμματικός έλεγχος, βοηθάει τον συντάκτη του κειμένου να εντοπίσει τα πιθανά λάθη και του υποδεικνύει τον γραμματικό κανόνα που ισχύει. Το συγκεκριμένο ηλεκτρονικό εργαλείο δεν μπορεί να έχει πρόσβαση στο νόημα του κειμένου, αλλά «συνεργάζεται» με τον συντάκτη, ο οποίος αποφασίζει για την τελική του μορφή.

«Μνημοσύνη»: Περιβάλλον Επεξεργασίας Συλλογών Κειμένων

Το περιβάλλον ΜΝΗΜΟΣΥΝΗ (Mnemosyne) αποτελεί ένα ολοκληρωμένο σύστημα επεξεργασίας φυσικής γλώσσας, το οποίο ενσωματώνει υψηλής ποιότητας γλωσσικούς πόρους και υπολογιστικά εργαλεία με στόχο την αυτόματη εξαγωγή δομημένης πληροφορίας από μη δομημένα ηλεκτρονικά έγγραφα. Χρησιμοποιείται κυρίως για την αυτόματη επεξεργασία ελεύθερων κειμενικών εγγράφων. Διασφαλίζει:

1. την επεξεργασία μεγάλου όγκου πληροφοριών,
2. υψηλή ακρίβεια στην αναγνώριση των ονοματικών οντοτήτων (Named Entities) και γεγονότων (Events),
3. δυνατότητα προσθήκης νέων πηγών δεδομένων με χαμηλό κόστος.

Κύρια χαρακτηριστικά

Δεδομένα εισόδου: Δυνατότητα επεξεργασίας συλλογών κειμένων, διαφόρων μορφών (HTML, PDF, TXT) και αποθηκευμένων σε ποικίλα μέσα (αρχεία, σελίδες ιστοτόπων, βάσεις δεδομένων).

Γλωσσικοί πόροι: Το σύστημα χρησιμοποιεί ποικίλους γλωσσικούς πόρους, όπως λεξιλόγια διαφόρων γλωσσών, ορθογραφικά λεξικά, μορφολογικά λεξικά, ορολογικά λεξικά, θησαυρούς κ.ά.

Επισημειώσεις: Επιτρέπει διαφορετικές αναλύσεις ενός κειμένου, οι οποίες εκφράζονται ως επισημειώσεις διαφορετικών επιπέδων. Η ύπαρξη διαφορετικών επιπέδων επισημειώσεων παρέχει μεγάλη ευελιξία στο σύστημα και δυνατότητα διασύνδεσης μεταξύ των αναλύσεων.

Αναλυτές και ροή παραγωγής: Οι αναλυτές αποτελούν τους μηχανισμούς παραγωγής των αναλύσεων. Συνδέονται μεταξύ τους μέσω διαφορετικών ροών παραγωγής, κατά τέτοιο τρόπο ώστε το αποτέλεσμα επεξεργασίας ενός αναλυτή να μπορεί να αποτελεί είσοδο για τον επόμενο. Οι ροές λειτουργούν παράλληλα επιτυγχάνοντας μεγαλύτερη ταχύτητα στην επεξεργασία.

Γλώσσα περιγραφής κανόνων: Η γλώσσα περιγραφής «Κανών» επιτρέπει τη δημιουργία σημασιολογικών επισημειώσεων που βασίζονται στον προσδιορισμό συντακτικών κανόνων.

Φιλτράρισμα: Τα αποτελέσματα μπορούν να φιλτράρονται προτού δοθούν προς επεξεργασία σε άλλους αναλυτές. Με τον τρόπο αυτό, εξασφαλίζεται η ελαχιστοποίηση της πληροφορίας που μεταφέρεται στην επόμενη φάση, απλοποιώντας κατά συνέπεια τους κανόνες των επόμενων αναλυτών.

Μηχανισμοί ασαφούς ταιριάσματος: Το σύστημα, αφού αναγνωρίσει μια ονοματισμένη οντότητα (π.χ. πρόσωπο, οργανισμό, εταιρεία κτλ.), παρέχει τη δυνατότητα ταιριάσματος της οντότητας αυτής με τις οντότητες που είναι καταχωρισμένες σε μια υπάρχουσα βάση δεδομένων. Οι μηχανισμοί που παρέχονται χωρίζονται σε δύο κατηγορίες: στους λεξικογραφικούς, που χρησιμοποιούν τεχνικές ορθογραφικής

διόρθωσης (π.χ. αποστάσεις λεκτικών), και στους στατιστικούς, που μετράνε την ομοιότητα του λεκτικού με τα λεκτικά της βάσης λαμβάνοντας υπόψη τον αριθμό και τη βαρύτητα των τριγραμμάτων (ή qγραμμάτων).

Εξαγωγείς: Εξειδικευμένοι αναλυτές αναλαμβάνουν τη μεταφορά της εξαγόμενης πληροφορίας στους επιθυμητούς προορισμούς και μορφές (π.χ. XML, Database tables κ.ά.).

Ειδικός μηχανισμός εποπτείας: Επιτρέπει την παρακολούθηση της όλης διαδικασίας, καθώς και την καταγραφή της σε διαφορετικά μέσα (αρχεία, βάση δεδομένων).

Έλεγχος εξαγόμενης πληροφορίας: Παρέχεται η δυνατότητα παρακολούθησης της εξαγόμενης πληροφορίας σε κάθε στάδιο της επεξεργασίας. Ο μηχανισμός αυτός επιτρέπει την αποσφαλματοποίηση της διαδικασίας βήμα προς βήμα.

Παραθυρική εφαρμογή για την επιβεβαίωση, διόρθωση και συμπλήρωση της εξαγόμενης πληροφορίας. Ο χρήστης της εφαρμογής μπορεί με εύχρηστο και ευέλικτο τρόπο να προσθέτει, να διορθώνει και να καταργεί επισημειώσεις της αυτόματης διαδικασίας, μεγιστοποιώντας έτσι την ποιότητα και την ακρίβεια της πληροφορίας.

Εφαρμογή ιστού για την παρουσίαση των αποτελεσμάτων με ποικίλους τρόπους, όπως έγχρωμες επισημειώσεις πάνω στο κείμενο, πίνακες ανά κατηγορία πληροφορίας, φιλτράρισμα σε όλα τα χαρακτηριστικά της πληροφορίας, συγκεντρωτικά αποτελέσματα και συγκρίσεις με αποτελέσματα άλλων μεθόδων.

2. Νέοι/Εμπλουτισμένοι πόροι

Οι πόροι που περιγράφηκαν παραπάνω, στο πλαίσιο του έργου εμπλουτίστηκαν, ενώ δημιουργήθηκαν και καινούργιοι. Ο εμπλουτισμός προήλθε κυρίως από την αποδελτίωση των σχολικών βιβλίων (που διαχειρίζεται και εκδίδει πλέον το ΙΤΥΕ), πληροφορία από τους συσσωρευτές «Φωτόδεντρο» και δευτερευόντως «Ιφιγένεια», αλλά και από το χαρακτηρισμό με επί πλέον πληροφορία του συνολικού περιεχομένου. Το αποτέλεσμα είναι η ύπαρξη ενός «καθολικού Λεξικού», αλλά και επί πλέον/βελτιωμένων πόρων που περιγράφονται στη συνέχεια. Επί πλέον, και πέρα από την «κλασσική» προσέγγιση της Επεξεργασίας Φυσικής Γλώσσας, στο πλαίσιο του έργου έγινε προσπάθεια αξιοποίησης και νεώτερων μεθόδων που χρησιμοποιούνται στην μηχανική μάθηση (machine learning) με σχετικούς στατιστικούς αλγορίθμους που εκπαιδεύονται μέσα από σώματα κειμένων.

Συνοπτικά τα νέα αλλά και βελτιωμένα συστατικά που εμπλούτισαν την εργαλειοθήκη μας παρουσιάζονται στις παρακάτω ενότητες.

Ταξινομία Γεωγραφικών όρων.

Στο παραδοτέο Π2.3, περιγράφηκε η Ταξινομία όρων Γεωγραφίας – Γεωλογίας. Ο πόρος αυτός είναι ξεχωριστός και σημαντικός για την περαιτέρω ανάπτυξη των υποδομών.

Γραμματικός Διορθωτής

Ο Γραμματικός διορθωτής που περιγράφηκε στο προηγούμενο κεφάλαιο, έδωσε δυο επιλογές/κατευθύνσεις στις απαιτήσεις/ανάγκες του έργου:

Αρχικά, το σύνολο των κανόνων που έχουν περιγραφεί στον διορθωτή, προσαρμόστηκαν και αξιοποιήθηκαν και για τις ανάγκες των παιχνιδιών του έργου. Πιο συγκεκριμένα, οι συγκεκριμένοι κανόνες (βλ. σχετικό Παράρτημα «Γραμματικός Διορθωτής»), αποτέλεσαν συμπλήρωμα των κατηγοριών λαθών που περιγράφηκαν στην Ενότητα Εργασίας 1.

Παράλληλα, δημιουργήθηκε μια έκδοση του γραμματικού διορθωτή, που εντοπίζει τα γραμματικά λάθη που έχουν σχέση με τα «τριγενή και δικατάληκτα» επίθετα και τον λανθασμένο σχηματισμό ή χρήση τους. Οι σχετικοί κανόνες που αξιοποιήθηκαν, καταγράφονται και αυτοί στο ίδιο παράρτημα.

Πιο συγκεκριμένα, έγινε προσπάθεια να διερευνηθεί το είδος των «λαθών» που απαντούν στην κλίση των τριγενών και δικατάληκτων επιθέτων (ης, -ης, -ες). Έγινε συστηματική περιγραφή των λαθών που γίνονται τόσο στην κλίση όσο και για τη λανθασμένη μορφολογική παραγωγή τους σε ονοματικές φράσεις. Η ομαδοποίηση των λαθών αυτών έγινε μέσα από σώματα κειμένων στο διαδίκτυο. Η συγκέντρωση των δεδομένων έγινε μέσω μηχανών αναζήτησης. Αναζητήθηκα οι ακολουθίες που αποτέλεσαν τις «λέξεις - κλειδιά» είτε έγινε η αναζήτηση βασισμένη σε συγκεκριμένες λανθασμένες συνάψεις, που επιλέχθηκαν με βάση τη σχετικά υψηλή συχνότητά τους σε σχέση με άλλες κατηγορίες λαθών. Με βάση αυτή την πληροφορία περιγράφηκαν σε κανόνες αυτές οι αποκλίσεις, με τη χρήση του Mnemosyne. Έγινε προσπάθεια ο σωστός σχηματισμός να μην περιγράφεται μόνο στο άμεσο γειτονικό γλωσσικό περιβάλλον (π.χ. άρθρο - επίθετο) αλλά και σε πιο σύνθετους σχηματισμούς (συμφωνία υποκειμένου με κατηγορούμενο).

Καθολικό Λεξικό

Η ξεχωριστή χρήση των λεξικών που χρησιμοποιούμε συνήθως στους επεξεργαστές κειμένου για να ελέγχουμε την ποιότητα του κειμένου που γράφουμε είναι αποδοτική χωρίς σημαντικές αδυναμίες. Για χρόνια η χρήση του διορθωτή κειμένου ήταν αποσπασμένη από την χρήση του γραμματικού διορθωτή και του θησαυρού. Αυτό έδινε την δυνατότητα στην εφαρμογή να φορτώνει μόνο ένα γλωσσικό πόρο

περιορίζοντας την κατανάλωση μνήμης που ήταν ένα από τα σοβαρά θέματα που είχε να αντιμετωπίσει. Με την μείωση του κόστους του hardware και την ταυτόχρονη αύξηση των επιδόσεων του, εμφανίστηκαν συνδυασμοί γλωσσικών πόρων με γνωστότερο την συνύπαρξη του γραμματικού και ορθογραφικού διορθωτή στον επεξεργαστή κειμένου της Microsoft. Η συνύπαρξη αυτή απαιτεί την ένωση των γλωσσικών πόρων (λεξικών) είτε σε φυσικό είτε σε λογικό επίπεδο.
Για την υποστήριξη των εκπαιδευτικών παιχνιδιών θέλουμε να αξιοποιήσουμε τους υπάρχοντες γλωσσικούς πόρους με μία κοινή προγραμματιστική διεπαφή (API – Application Programming Interface). Οι τρεις κύριοι γλωσσικοί πόροι που ενοποιήθηκαν κατά στην διαδικασία αυτή ήταν το μορφολογικό λεξικό, ο θησαυρός συνωνύμων και η βάση του «Χτίζω Λέξεις». Η ένωση έγινε με δομές δεδομένων στην μνήμη όπου φορτώνονται τα τρία λεξικά και συμπληρώνεται η πληροφορία των δομών ή δημιουργούνται οι αναφορές. Στην συνέχεια το ενοποιημένο λεξικό αξιοποιείται σε αναζητήσεις αλλά και εξαγωγή σύνθετης πληροφορίας.

Πολυμορφικό Λεξικό – CompressedTrie

Στην Επεξεργασία Φυσικής Γλώσσας συμμετέχουν δύο συστατικά, οι γλωσσικοί πόροι και οι τεχνολογίες (μηχανισμοί) επεξεργασίας. Τα δύο συστατικά είναι στενά συνδεδεμένα, οι τεχνολογίες χρειάζονται τους γλωσσικούς πόρους για να κάνουν την δουλειά τους. Οι γλωσσικοί πόροι ενσωματώνουν την γνώση που έχουμε για μία πτυχή της γλώσσας. Το λεξικό ορθογραφίας περιέχει την σωστή ορθογραφία των λέξεων, το λεξικό συλλαβισμού έχει τις «δύσκολες» περιπτώσεις συλλαβισμού, ο θησαυρός τις σημασίες τα συνώνυμα και αντίθετα, κ.λπ. Από τα χαρακτηριστικά που πρέπει να έχουν τα λεξικά είναι:

1. Συμπιεσμένη μορφή για καλύτερη εκμετάλλευση του χώρου. Η αρχική αναπαράσταση των γλωσσικών πόρων είναι σε μία μορφή κειμένου και σε κάποιο φορμαλισμό που έχει την δυνατότητα να εκφράσει όλες τις ιδιότητες που φέρουν από το γλωσσικό μέρος.
2. Να συμπεριλαμβάνει τα κατάλληλα ευρετήρια και δομές αποθήκευσης των δεδομένων για αποδοτική προσπέλαση της πληροφορίας τους. Το φόρτωμα του λεξικού στην μνήμη από τους μηχανισμούς που το έχουν ανάγκη πρέπει να είναι γρήγορο. Οι δομές των δεδομένων που χρησιμοποιεί το λεξικό βελτιώνουν λειτουργίες αναζήτησης που χρειάζεται ο μηχανισμός για την αποδοτική του λειτουργία.

Το πολυμορφικό λεξικό έχει τα παραπάνω χαρακτηριστικά και χρησιμοποιείται σε αρκετούς γλωσσικούς μηχανισμούς. Η δομή του αρχείου αποτελείται από τα παρακάτω μέρη.

1. Πρώτη επικεφαλίδα με τις ιδιότητες του λεξικού. Οι ιδιότητες αναφέρονται στον τύπο του λεξικού και βοηθούν στην ερμηνεία του περιεχομένου του. Μπορεί να απουσιάζει κυρίως στις πρώτες εκδόσεις του πολυμορφικού λεξικού.
2. Ευρετήριο λέξεων σε δομή `CompressedTrie`. Όλα τα πολυμορφικά λεξικά προσφέρουν πρόσβαση στα δεδομένα τους μέσα από ένα καθορισμένο λεξιλόγιο. Το ευρετήριο είναι μία συνάρτηση που απεικονίζει μία λέξη σε ένα αριθμό. Ο αριθμός αυτός χρησιμοποιείται στην συνέχεια ως αναγνωριστικό της πληροφορίας που αποθηκεύουν οι πίνακες που περιγράφονται στο επόμενο στοιχείο.
3. Δεύτερη επικεφαλίδα με ιδιότητες του λεξικού με έμφαση τις ιδιότητες των πινάκων δεδομένων. Η επικεφαλίδα αυτή εμφανίζεται και στις παλαιότερες εκδόσεις του λεξικού.
4. Πίνακες δεδομένων εξειδικευμένα για κάθε διαφορετικό τύπο λεξικού. Μετά την μετατροπή μίας λέξης σε αριθμό, οι προσπέλαση των δεδομένων που σχετίζονται με την λέξη άμεσα ή έμμεσα, γίνεται με τους πίνακες δεδομένων. Οι πίνακες αυτοί υλοποιούν την απεικόνιση από τον αριθμό που αντιστοιχεί στην εισόδιο λέξη προς την κεντρική πληροφορία του λεξικού.

Ένα από τα πολυμορφικά λεξικά με πολλές εφαρμογές είναι το λεξικό συχνοτήτων. Χρησιμοποιείται από πολλούς αλγόριθμους μηχανικής μάθησης όπως TF-IDF, BM25, K-Means, κ.α. Θα δούμε την δομή του λεξικού ως παράδειγμα πολυμορφικού λεξικού. Τα δεδομένα προκύπτουν από την επεξεργασία κειμένων που ονομάζονται έγγραφα και θεωρούνται δοχεία λεκτικών μονάδων. Στόχος μας είναι να ευρετηριάσουμε τα έγγραφα αυτά ώστε να μπορεί να γίνει μία στοχαστική αναζήτηση με βάση λεκτικές μονάδες που περιέχουν. Κατά συνέπεια το κάθε έγγραφο πρέπει να έχει ένα μοναδικό αναγνωριστικό που είναι ένα κορδόνι χαρακτήρων. Παράδειγμα τέτοιας αναζήτησης είναι και το «google» που με βάση λέξεις που δίνουμε μας επιστρέφει τις «καλύτερες» σελίδες. Η λεκτική μονάδα μπορεί να είναι τα στοιχεία που θα μας επιστρέψει ένας στοιχειοποιητής (tokenizer) ή κομμάτια στοιχείων για την περίπτωση των q-grams. Το q-gram εκφράζει γενικά μια σειρά μεγέθους q από συστατικά. Τα συστατικά σε κάποιες εφαρμογές είναι λέξεις ή σημεία, σε άλλες μπορεί να είναι οντότητες και σε άλλες όπως εδώ χαρακτήρες. Στην περίπτωση που το q = 0, τότε η μονάδα είναι η λέξη. Αν για παράδειγμα το q = 3 τότε η μονάδα είναι τριγράμματα. Για παράδειγμα η λέξη ‘καλά’ αποτελεί μία λεκτική μονάδα αν το q = 0 και σπάει στα τριγράμματα (‘ κ’, ‘ κα’, ‘καλ’, ‘αλά’, ‘λά ’, ‘ά ’) αν το q = 3. Τα δεδομένα που αποθηκεύει το λεξικό συχνοτήτων είναι:

1. Δεν έχουμε πρώτη επικεφαλίδα.
2. Ευρετήριο λέξεων σε δομή CompressedTrie. Για κάθε λεκτική μονάδα το ευρετήριο δίνει έναν αριθμό.
3. Αποθήκευση του παράγοντα q που καθορίζει το είδος της λεκτικής μονάδας. Αποθήκευση του μέσου όρου των λεκτικών μονάδων ανά κείμενο.
4. Πίνακας συχνοτήτων λεκτικών μονάδων (ή q-grams). Είναι ένας πίνακας αριθμών που για κάθε λεκτική μονάδα μας λέει την συχνότητα εμφάνισης στα έγγραφα της συλλογής από την οποία κτίσαμε το λεξικό.
5. Πίνακας εγγράφων. Για κάθε έγγραφο της συλλογής γράφουμε συγκεντρωτικά πληροφορίες όπως:
    a. Το αναγνωριστικό του εγγράφου.
    b. Πίνακα λεκτικών μονάδων όπου κάθε γραμμή του πίνακα έχει δύο αριθμούς. Ο πρώτος είναι η λεκτική μονάδα και ο δεύτερος πόσες φορές εμφανίστηκε στο έγγραφο η συγκεκριμένη λεκτική μονάδα.

Όπως αναφέραμε παραπάνω η πληροφορία που έχει το συγκεκριμένο λεξικό μας επιτρέπει να υλοποιήσουμε έναν αριθμό από αλγόριθμους μηχανικής μάθησης. Παρακάτω θα αναφερθούμε στον αλγόριθμο TF-IDF έναν από τους σημαντικότερους στο χώρο της Ανάκτησης Πληροφορίας (Information Retrieval).

Εργαλεία κατασκευής λεξικών

Εκτός από το Καθολικό Λεξικό που είδαμε στην προηγούμενη ενότητα, το οποίο κατασκευάζεται δυναμικά στην μνήμη από σύνθεση λεξικών, τα λεξικά που χρησιμοποιούνται στις αναλύσεις είναι δυαδικά (binary) αρχεία με συμπιεσμένη την πληροφορία με κατάλληλα ευρετήρια για την αποδοτική τους χρήση. Όλα τα λεξικά που χρησιμοποιούμε έχουν κτιστεί με ξεχωριστά εργαλεία του Mnemosyne. Το Mnemosyne διαθέτει ξεχωριστά εργαλεία για την κατασκευή λεξικών διαφορετικών τύπων. Τα εργαλεία κατασκευής λεξικών και οι τύποι λεξικών που χρησιμοποιούνται είναι:

- Λεξικά ορθογράφου με το εργαλείο `MakeDAWG`. Το εργαλείο αυτό κατασκευάζει μια δομή αυτομάτου γνωστό με το όνομα Directed Acyclic Word Graph (DAWG).

- Λεξικά θησαυρού με το εργαλείο ThesCreator. Η δομή που χρησιμοποιείται για τον θησαυρό είναι το Trie[1] και συγκεκριμένα μια παραλλαγή γνωστή με το όνομα Compressed Trie. Η δομή CompressedTrie χρησιμοποιείται ως ευρετήριο και σε άλλους τύπους λεξικών.
- Λεξικά χαρακτήρων & συλλαβών (locale) με πληροφορία για τους χαρακτήρες που θα εμφανιστούν στα κείμενα, τα φωνήματα και τις ισοδυναμίες τους. Το εργαλείο που χρησιμοποιούμε είναι το CompileLocale και το λεξικό περιέχει πίνακες χαρακτήρων, φωνημάτων και αυτόματα ισοδυναμιών για την υποβοήθηση της λειτουργίας της διόρθωσης.
- Λεξικά συλλαβισμού που περιέχουν λέξεις εξαιρέσεις που δεν συλλαβίζονται σωστά με τους κανόνες συλλαβισμού. Χρησιμοποιούμε με το εργαλείο MakeDAWG.
- Λεξικά Μορφολογίας με το εργαλείο MorphoDictionary. Τα λεξικά μορφολογίας στηρίζονται στην δομή CompressedTrie για να δημιουργήσουν ένα ευρετήριο από τις λέξεις προς τα μορφολογικά χαρακτηριστικά που έχουν σε κάθε λήμμα.
- Λεξικά Εγκυκλοπαίδειας με το εργαλείο OmnibusCreator. Τα λεξικά αυτά είναι επεκτάσεις των μορφολογικών λεξικών. Χρησιμοποιούν την δομή CompressedTrie για να δημιουργήσουν ένα ευρετήριο από τις λέξεις προς τα λήμματα τα οποία ενσωματώνουν μορφολογική και σημασιολογική πληροφορία. Τα ευρετήρια συμπεριλαμβάνουν όλες τις κλιτικές μορφές αλλά και άλλες μορφολογικές παραλλαγές των λέξεων που μπορεί να δείχνουν σε ένα λήμμα.
- Λεξικά όρων και πολυλεκτικών μονάδων. Οι λεκτικές μονάδες μπορεί να είναι λέξεις, τεχνικές κατασκευές (ψευδολέξεις) και κομμάτια λέξεων (3-γράμματα, 4-γράμματα, κ.λπ.). Χρησιμοποιούμε το εργαλείο DataCompiler που όπως θα δούμε παρακάτω έχει αρκετές λειτουργίες για την επεξεργασία του υλικού που θα χρησιμοποιηθεί για την κατασκευή των λεξικών.
- Λεξικά κανόνων τα οποία δημιουργούνται από το εργαλείο KanonCompiler. Τα λεξικά αυτά χρησιμοποιούνται στην ανάλυση κειμένων και έχουν ως βάση ευρετήρια συμβόλων πάνω στην δομή CompressedTrie.
- Λεξικά μηχανικής μάθησης που αποθηκεύουν μοντέλα που δημιουργήθηκαν κατά την εκπαίδευση ενός αλγόριθμου μηχανικής μάθησης. Θα δούμε σε λεπτομέρεια κάποια από τα μοντέλα αυτά στις επόμενες παραγράφους. Τα λεξικά αυτά κτίζονται με τα εργαλεία DataCompiler και DataTrainer και βασίζονται στην δομή CompressedTrie για την ευρετηρίαση των λέξεων ή λεκτικών μονάδων που σχηματίζονται από χαρακτήρες.
- Πολυλεξικό ή έγγραφο λεξικών (Lexicon Document), ένα σύνθετο λεξικό δοχείο άλλων λεξικών. Το σύνθετο λεξικό ενσωματώνει έναν αριθμό από λεξικά που είδαμε στις προηγούμενες παραγράφους με στόχο να καλύψει μια σύνθετη λειτουργικότητα. Για παράδειγμα ο ορθογραφικός διορθωτής χρησιμοποιεί λεξικά συντμήσεων, κύριων ονομάτων, λεξικά χαρακτήρων και το κυρίως λεξικό των λέξεων του μορφολογικού λεξικού της Neurolingo. Όλα αυτά τοποθετούνται σε ένα πολυλεξικό και χρησιμοποιούνται από τον ορθογραφικό διορθωτή. Το εργαλείο κατασκευή του πολυλεξικού είναι το LexiconDoc.

Εκτός από τα εργαλεία κατασκευής έχουμε ένα σύνολο βοηθητικών εργαλείων για την προετοιμασία της εισόδου πριν την επεξεργασία τους από τους κατασκευαστές λεξικών. Μερικά από αυτά είναι:

- SortFile: Ταξινομεί τα στοιχεία ενός αρχείου σε αύξουσα λεξικογραφική σειρά.
- SortFileUsingLocale: Παρόμοια με το SortFile με την διαφορά ότι η λεξικογραφική σειρά καθορίζεται από ένα λεξικό χαρακτήρων που δίνεται στο εργαλείο.

[1] https://en.wikipedia.org/wiki/Trie

- ConcatenateFiles: Ενοποιεί δύο ή περισσότερα αρχεία σε ένα
- DataPrep: Καθαρίζει και προετοιμάζει τα δεδομένα για επεξεργασία. Θα το παρουσιάσουμε σε μεγαλύτερο βάθος παρακάτω
- Word2Vec: Είναι ένα κάλυμμα για το αντίστοιχο πρόγραμμα (Word2vec[2]). Δέχεται ένα αρχείο κειμένου και παράγει τις **ενσωματώσεις λέξεων** (**word embeddings**), δηλ. τα ανύσματα λέξεων. Στην συνέχεια ο DataCompiler θα χρησιμοποιήσει την έξοδο αυτή για να δημιουργήσει το αντίστοιχο λεξικό.
- GloVe: Αντίστοιχο κάλυμμα για το πρόγραμμα[3] το οποίο δημιουργεί και αυτό ενσωματώσεις λέξεων.
- DataTrainer: Ένα εργαλείο που χρησιμοποιείται ως εναλλακτικός τρόπος δημιουργίας μοντέλων **λογιστικής παλινδρόμησης** χρησιμοποιώντας ως είσοδο ένα λεξικό που δημιουργήθηκε σε προηγούμενη φάση και όχι το αρχικό κείμενο.

Λεξικά Μηχανικής Μάθησης

Τα λεξικά Μηχανικής Μάθησης αποθηκεύουν μοντέλα που εκπαιδεύτηκαν με την επεξεργασία μιας συλλογής και θα χρησιμοποιηθούν στην ανάλυση κειμένων. Θα δούμε τα εργαλεία αυτά σε μεγαλύτερο βάθος μιας και συμμετέχουν στις νέες τεχνολογίες που ενσωμάτωσε το έργο μέσω του Mnemosyne.

- Το εργαλείο `DataPrep` χρησιμοποιείται στην προετοιμασία συλλογής από αρχεία κειμένων με στόχο την επεξεργασία τους από αλγόριθμους μηχανικής μάθησης. Η κύρια δουλειά του είναι να εξάγει το καθαρό κείμενο απαλλαγμένο από άγνωστους χαρακτήρες, τεχνικές λέξεις, αναφορές, μετα-χαρακτηριστικά και τυπογραφικές ή στιλιστικές προσμίξεις. Οι συλλογές κειμένων που βρίσκουμε στον παγκόσμιο ιστό χρησιμοποιούν πολλές μορφές ηλεκτρονικών αρχείων, όπως:
    - .txt: αρχεία με καθαρό κείμενο χωρίς διαμόρφωση.
    - .htm, .html αρχεία με διαμόρφωση για να παρουσιαστούν σε ένα φυλλομετρητή ιστού (web browser).
    - .csv (comma separated values), αρχεία κειμένου όπου κρατάνε εγγραφές των οποίων τα πεδία χωρίζονται με ένα χαρακτήρα διαχειριστή που συνήθως είναι το κόμμα από όπου πήραν και το όνομα. Ένας άλλος χαρακτήρας που χρησιμοποιείται συχνά ως διαχωριστής είναι ο tab (‘\t’). Από τα πεδία αυτά κάποια έχουν κείμενο που θέλουμε να εξάγουμε για επεξεργασία και κάποια πεδία είναι άσχετα. Θα δούμε παρακάτω πως επιλέγουμε τα πεδία επεξεργασίας.
    - .pdf αρχεία που χρησιμοποιούνται συνήθως για βιβλία και άρθρα που έχουν εκδοθεί και πιθανό εκτυπωθεί. Παρουσιάζουν το περιεχόμενο στην μορφή που θα είχε όταν εκτυπωθεί.
    - .xml αρχεία τα οποία έχουν την δυνατότητα να αποτυπώσουν σύνθετες δομές. Η μορφή αυτή παραμένει δημοφιλής επειδή καλύπτει πολλαπλές ανάγκες αναπαράστασης περιεχομένου, μεταδεδομένων, στυλ εμφάνισης, κ.α. Το μειονέκτημα τους είναι η επιβάρυνση που προσθέτουν στο περιεχόμενο και στην επεξεργασία τους. Από τα .xml αρχεία θα πρέπει να γνωρίζουμε τα στοιχεία που περιέχουν χρήσιμη πληροφορία για να την εξάγουμε. Παρακάτω θα δούμε πως γίνεται αυτή η επιλογή.
    - .json αρχεία που χρησιμοποιήθηκαν ως μια αποδοτικότερη αντικατάσταση του .xml για την ανταλλαγή μηνυμάτων μεταξύ υπολογιστών στο internet.

    Οι επιλογές που υπάρχουν στην εκτέλεση του εργαλείου `DataPrep` όπως εκτυπώνονται στην κονσόλα παρουσιάζονται παρακάτω.

[2] https://en.wikipedia.org/wiki/Word2vec
[3] https://en.wikipedia.org/wiki/GloVe_(machine_learning)

```
      Usage: DataPrep <options> (<terms_def.xml> |<terms_def.txt> | <terms_def.csv>
      |
                                 <terms_def.tbx | directory>)+
      (--help|-h)                                    : Print this command and options
                                                       info.
      (--output_directory|-od) <output directory>.   : The folder where the prepared
files
                                                       will be stored.
      (--output_file|-of) <output file>              : The path of output file. Input
                                                       files will be filtered and
                                                       concatenated in this file.
      (--name_pattern|-np) <pattern>                 : The name pattern (e.g. *.csv)
for
                                                       the files in folders.
      (--replace_output|-ro)                         : Replace output file, i.e.
truncate
                                                       the output file.
      (--ignore_header|-ih)                          : Ignore (does not export) the
first
                                                       (header) line.
      (--ignore_header_but_one|-iho)                 : Ignore (does not export) the
first
                                                       (header) line of all files but
the
                                                       first one.
      (--ignore_line_pattern|-ilp) <pattern>         : Ignore (does not export) lines
the
                                                       match the pattern. May
repeated.
      (--max_word|-mw) <length>                      : Maximum length of a valid
word.
      (--with_delimiters|-wd                         : Split words and delimiters,
keep
                                                       delimiters for training.
      (--locale_dict|-ld) <locale_dictionary.dic>    : Locale dictionary file defines
the
                                                       language of the dataset.
      (--normalize|-nr)                              : Normalize the text.
      (--min_num_words_per_line|-mnwl) <num>         : Specifies the minimum number
of
                                                       words in a valid line (default
1).
      (--percentage_of_locale|-pol) <percentage>     : Percentage of characters in a
line
                                                       that is in locale in order to
be
                                                       exported.
      (--ignore_patterns|-ip)  <expr file>           : The <expr file> contains an
ignore
                                                       exression per line.
      (--file_type|-ft) <type>                       : The file type. Possible values
                                                       ('csv', 'txt', 'html', 'xml',
'pdf',
                                                       'json').
```

```
      (--traverse_subfolders|-ts)                    : Input directories and
                                                       subdirectories will traversed
to
                                                       collect the appropriate files.
      (--csv_id_column|-cid) <id column>             : For 'csv' file type, this is
the id
                                                       or class column.
      (--csv_data_columns|-cdt) <data columns>       : For 'csv' file type, these are
the
                                                       data columns comma separated.
      (--csv_use_tab_sep|-cut)                       : Use tab character as csv
separator.
      (--csv_html_content|-chc)                      : Process the content as html,
i.e.
                                                       clear the html tags.
      (--csv_separator|-csep) <separator>            : Define the separator in csv
files
                                                       if no the tab character.

Default is
                                                       ','
      (--csv_data_only_export|-cdex)                 : Export only the data columns.
      (--xml_data_only_export|-xdex)                 : Export only the data
(characters)
                                                       part of the element.
      (--xml_data_elements|-xdels) <xml elements>    : Use only the specified
elements to
                                                       get their content.
      (--json_data_fields|-jsdf) <json fields>       : Use only the specified field
to get
                                                       their content and concatenate

them
                                                       in a text.
      (--json_array|-jsar)                           : JSon file contains an array of
                                                       records. Merge all the
interested
                                                       fields.
      (--decode-charset|-dcs) <charset map json>     : Use the charset mapping to
decode
                                                       the contents.
```

Οι υποχρεωτικές παράμετροι του προγράμματος είναι να προσδιορίσουμε την είσοδο και την έξοδο της επεξεργασίας. Ως είσοδο μπορούμε να έχουμε ένα ή περισσότερα αρχεία και φακέλους. Ως έξοδο μπορούμε να καθορίσουμε ένα φάκελο (`--output_directory`) στον οποίο θα δημιουργηθούν τα επεξεργασμένα αρχεία εισόδου ή το όνομα ενός αρχείου (`--output_file`) όπου θα δημιουργηθεί και θα ενσωματώσει τα επεξεργασμένα περιεχόμενα όλων των αρχείων εισόδου. Οι επιλογές που καθορίζουν τις λεπτομέρειες λειτουργίας του προγράμματος είναι:

- `--name_pattern` καθορίζει την μορφή του ονόματος των αρχείων που θα χρησιμοποιήσουμε στην περίπτωση που η είσοδος είναι φάκελος και μπορεί να έχει και άσχετα αρχεία.
- `--replace_output` στην περίπτωση που τα αρχεία εξόδου υπάρχουν τα περιεχόμενά τους θα σβήσουν. Αν δεν καθοριστεί η επιλογή αυτή και κάποιο αρχείο εξόδου υπάρχει ήδη θα παραμείνει αμετάβλητο και θα παραχθεί ένα μήνυμα λάθους.
- `--ignore_header` στην περίπτωση που η είσοδος είναι .csv αρχεία θα αγνοηθεί η πρώτη γραμμή γιατί περιέχει μια επικεφαλίδα και όχι δεδομένα.

- `--ignore_header_but_one` παρόμοια με την προηγούμενη επιλογή με την διαφορά ότι θα αγνοήσει την πρώτη γραμμή σε όλα τα .csv αρχεία εκτός του πρώτου. Αυτό θα έχει σαν συνέπεια να βγει στην έξοδο η επικεφαλίδα μία φορά μόνο και όχι για κάθε αρχείο. Η επιλογή χρησιμοποιείται όταν επεξεργαζόμαστε πολλά .csv αρχεία και ενώνουμε το περιεχόμενο σε ένα αρχείο εξόδου. Στην περίπτωση αυτή δεν θέλουμε να βγάζουμε στην έξοδο την επικεφαλίδα για κάθε αρχείο που επεξεργαζόμαστε αλλά μόνο για το πρώτο.
- `--ignore_line_pattern` δέχεται μια κανονική έκφραση (regular expression) ως όρισμα και τσεκάρει μία μία τις γραμμές ενός αρχείου εισόδου αν ικανοποιούν την έκφραση. Αν η έκφραση ικανοποιείται τότε η γραμμή αγνοείται και η επεξεργασία συνεχίζει με την επόμενη γραμμή. Η λειτουργία αυτή είναι χρήσιμη σε περιπτώσεις που τα αρχεία εισόδου περιέχουν και επιπλέον δεδομένα εκτός του κειμένου (στιλιστικές οδηγίες, μεταδεδομένα, κ.λπ.).
- `--max_word` καθορίζει το μέγιστο μήκος που μπορεί να έχει μία λέξη. Αν μία λεκτική μονάδα που αποτελείται από έγκυρα γράμματα έχει μεγαλύτερο μήκος από αυτό θα αγνοηθεί.
- `--with_delimiters` μας λέει ότι οι διαχωριστές των λέξεων, όπως π.χ. το κόμμα, θα διατηρηθούν στην έξοδο. Συνήθως οι διαχωριστές δεν βγαίνουν στην έξοδο και δεν συμμετέχουν στην εκπαίδευση κάποιου μοντέλου.
- `--locale_dict` καθορίζει το αρχείο χαρακτήρων που ορίζει τους έγκυρους χαρακτήρες που θα έχουμε στην είσοδο και τις ιδιότητές τους.
- `--normalize` μας λέει ότι η έξοδος θα είναι μια κανονικοποιημένη μορφή της εισόδου. Η κανονικοποίηση για τα ελληνικά σημαίνει αναγωγή όλων των χαρακτήρων σε κεφαλαίους χωρίς τους τόνους.
- `--min_num_words_per_line` καθορίζει άλλο ένα φίλτρο για την απομάκρυνση άσχετων γραμμών από ένα αρχείο εισόδου. Εδώ κοιτάμε τον αριθμό των λέξεων που έχει μία γραμμή και αν είναι μικρότερος από τον καθορισμένο εδώ τότε η γραμμή αγνοείται.
- `--percentage_of_locale` άλλο ένα φίλτρο γραμμών στα αρχεία εισόδου. Για κάθε γραμμή κοιτάμε το ποσοστό των έγκυρων χαρακτήρων και αν το ποσοστό είναι μικρότερο από το καθορισμένο εδώ τότε η γραμμή αγνοείται.
- `--ignore_patterns` παρόμοιο με το `--ignore_line_pattern` με την διαφορά ότι εδώ καθορίζουμε ένα αρχείο που περιέχει τις εκφράσεις, μία σε κάθε γραμμή. Χρησιμοποιούμε το αρχείο όταν έχουμε πολλές εκφράσεις και δεν θέλουμε να μεγαλώσουμε πολύ το μήκος της γραμμής εκτέλεσης.
- `--file_type` παρόμοιο με την επιλογή `--name_pattern` με την διαφορά ότι εδώ ο έλεγχος είναι απλούστερος και κοιτά μόνο την κατάληξη του ονόματος του αρχείου για να προσδιοριστεί ο τύπος του.
- `--traverse_subfolders` στην περίπτωση που έχουμε προσδιορίσει κάποιο φάκελο εισόδου τότε αυτή η επιλογή μας λέει ότι θα πρέπει να συνεχίσουμε την αναζήτηση αρχείων και στους υποφακέλους του.
- `--csv_id_column` στην περίπτωση επεξεργασίας .csv αρχείων, η επιλογή αυτή καθορίζει την στήλη που περιέχει το αναγνωριστικό (κλειδί) της εγγραφής.
- `--csv_id_column` στην περίπτωση που έχουμε ένα πεδίο ενδιαφέροντος κειμένου σε .csv αρχεία, η επιλογή αυτή μας επιτρέπει να καθορίσουμε ποιο είναι το πεδίο αυτό (αριθμός στήλης).
- `--csv_data_columns` στην περίπτωση που έχουμε περισσότερα από ένα πεδία ενδιαφέροντος σε .csv αρχεία μπορούμε να χρησιμοποιήσουμε αυτή την επιλογή και να

καθορίσουμε τις στήλες που θα συμμετέχουν στην επεξεργασία. Οι αριθμοί των στηλών θα διαχωρίζονται με κόμμα (π.χ. 2,3,5).

- `--csv_use_tab_sep` για την περίπτωση των .csv αρχείων καθορίζει ότι ο διαχωριστής δεν είναι το κόμμα (‘,’) αλλά ο χαρακτήρας tab (‘\t’).
- `--csv_html_content` καθορίζει ότι τα πεδία δεδομένων στο .csv αρχείο περιέχουν .html κείμενο και πρέπει να καθαρίσει με το αντίστοιχο φίλτρο.
- `--csv_separator` καθορίζει τον διαχωριστή στα .csv αρχεία. Χρησιμοποιείται για την περίπτωση που ο διαχωριστής δεν είναι το κόμμα και ο tab αλλά κάποιος άλλος (π.χ. ‘;’).
- `--csv_data_only_export` και εδώ για την περίπτωση των .csv αρχείων η επιλογή αυτή μας λέει στην έξοδο θα βγάλουμε μόνο τα πεδία δεδομένων και όχι άλλα πεδία συμπεριλαμβανομένου του αναγνωριστικού.
- `--xml_data_only_export` παρόμοια με την προηγούμενη επιλογή και για την περίπτωση που η είσοδος είναι σε μορφή .xml, η επιλογή αυτή μας λέει ότι θα εξάγουμε μόνο τα στοιχεία δεδομένων.
- `--xml_data_elements` καθορίζει τα στοιχεία xml που περιέχουν χρήσιμο κείμενο για επεξεργασία. Ο αναγνώστης xml που χρησιμοποιούμε μπορεί να επεξεργαστεί σύνθετες xml δομές με μεγάλο βάθος αλλά συνήθως τα αρχεία xml που έχουν συλλογή κειμένων έχουν ρηχές δομές και τα στοιχεία που έχουν χρήσιμη πληροφορία είναι στο ίδιο επίπεδο από την κορυφή (ρίζα). Τα ονόματα των στοιχείων αυτών χωρισμένα με κόμμα είναι παράμετροι αυτής της επιλογής.
- `--json_data_fields` παρόμοια με την προηγούμενη επιλογή για την περίπτωση που έχουμε .json αρχεία.
- `--json_array` για την περίπτωση που τα πεδία κειμένου είναι φωλιασμένα σε μία διάταξη (array) η επιλογή αυτή μας λέει ότι θα πρέπει να γίνει συνένωση των στοιχείων της διάταξης πριν βγουν στην έξοδο ως ένα κείμενο.
- `--decode-charset` για την περίπτωση που η είσοδος είναι κωδικοποιημένη. Η παράμετρος της επιλογής είναι ένα Json αρχείο που περιέχει την απεικόνιση των χαρακτήρων. Διαχειριζόμαστε απλές κωδικοποιήσεις όπου ένας χαρακτήρας έχει απεικονιστεί σε κάποιον άλλο (π.χ. ο ΄α΄ ➔ ‘8’).

- Το εργαλείο `DataCompiler` χρησιμοποιείται για την κατασκευή λεξικών που συμμετέχουν στις αναλύσεις κειμένων. Ο κύριος στόχος για την δημιουργία του ήταν να κτίζουμε λεξικά από δεδομένα που υπάρχουν σε βάσεις δεδομένων στις εγκαταστάσεις του Mnemosyne ή συλλογές που θα καταβιβάζουμε από το διαδίκτυο. Στις περιπτώσεις αυτές τα δεδομένα δεν είναι οργανωμένα σε βολικές μορφές όπως τα δεδομένα των γλωσσικών μας πόρων. Το εργαλείο πρέπει να αναγνωρίζει αρκετές πηγές, να μπορεί να αναλύσει το περιεχόμενό τους, να εξάγει την χρήσιμη πληροφορία και να κτίσει τα κατάλληλα λεξικά. Ένα μέρος της λειτουργίας του `DataPrep` που αφορά την διαχείριση των πηγών και των μορφών δεδομένων τους έχει προέλθει από τον `DataCompiler`. Σε όλες τις περιπτώσεις των λεξικών που δημιουργεί ο DataCompiler έχουμε ένα ή περισσότερα ευρετήρια σε δομή CompressedTrie ακολουθούμενα από τα δεδομένα ή μοντέλα που λεξικού. Ο `DataCompiler` δέχεται τις παρακάτω επιλογές από την γραμμή εκτέλεσης του.

```
Usage: Compiler <options> (<terms_def.xml> | <terms_def.txt> | <terms_def.csv> |
                           <terms_def.tbx>)+
```

```
(--morpho_dict|-md)  <morpho_dictionary.dic>
(--thes_dict|-thd)   <thesaurus_dictionary.dic>
(--locale_dict|-ld)  <locale_dictionary.dic>
(--cust_lexicon|-cl) <cust_lexicon.xml>
(--freqs_file|-ff)   <freq_file>
(--dump_qfile|-dq)   <qgrams_file>
(--dump_rfile|-dr    <results_file>)
(--dictionary|-dict) <dictionary.dic> : the produced dictionary
(--morpho_lemma|-ml)
(--params|-p)        <params>
(--compute_freq|-cf)
(--dump_ids|-di)
(--dump_freqs|-df)
(--word_index|-wi)   <index_num>
(--create_text|-ct)
(--db_index|-db
(--normalized|-nr)
(--lemmatized|-lz)
(--tokenization_method|-tm)  <tokenization method>
(--normalization_method|-nm) <normalization method>
(--use_only_one_lemma|-uol)
(--replace_with_lemma|-rl)
(--use_stem_on_unknown|-usu)
(--use_locale_norma|-uln)
(--fill_morphology|-fmr)
(--expand|-ex)
(--expand_with_lemmas|-exl)
(--replace|-rp)
(--single|-sn)
(--combine|-cb)
(--concatenate|-cn)
(--use_term_stats|-uts)
(--dump_only|-do)
(--dump_qgrams|-dqg)
(--dump_results|-drs)
(--dump_sep|-sep <dump_sep>)
(--qgrams|-qg) <q>
(--auto_id|-ida)
(--id_column|-idc)       <column>
(--data_delimiter|-dtdl) <delimiter>
(--data_columns|-dtcs)   <comma separated columns>
(--num_columns|-nc)      <#cols>
(--sample_count|-sc)     <#recs>
(--no_sep|-nos
(--tab_sep|-ts
(--num_sep|-ns
(--sep_expr|-se)         <term_seperation_expr>
(--skip_header_line|-shl)
(--skip_double_entries|-sde)
(--use_disk_structs|-ds)
(--use_memory_structs|-ms)  : This is the default option
(--tmp_folder|-tf) <tmp_folder>)
(--bayes_dict|-bd)      : Builds a Naive Bayes dictionary from train data.
(--regression_dict|-rd) : Builds a Multinomial Logistic Regression dictionary
                          from train data.
```

```
      (--compiled_dic|-cd)    : Compiles the dataset into a binary dictionary format
                                indexed by words (tokens).
      (--tfidf_dict|-idf)     : builds the tfidf/bm25 terms dictionary
      (--inverted_index|-in)  : builds the inverted (key -> {word}+) dictionary
      (--parallel_train|-pt)                : Parallel training of Multinomial
Logistic
                                              Regression model dictionary.
      (--learning_rate|-lr)                 : Regression learning rate.
      (--batch_size|-bs)                    : Regression mini batch size.
      (--epoch_stuck_count|-esc) <count>    : Regression training 'max number of
epochs
                                              without progress'
      (--missed_percentage|-mp) <percentage> : Regression training, accepted
percentage
                                              of missed documents/terms
      (--print_train_info|-pti)             : Regression print training information
                                              Creates a terminology dictionary from
a
                                              text or xml term's definition.
                                              It uses the morphological dictionary
                                              to find the lemmas.
      (--bigdata|-bdt)              : Big dataset, use disk temporary files to
store the
                                      data.
      (--export_word_refs|-ewr)     : Export the word references, i.e. the
documents
                                      that each word appears in.
      (--word_vector|-wv)           : Stores a word vector binary dictionary.
      (--word2vec|-w2v)             : Input is a word2vec vector file.
      (--glove|-gv)                 : Input is a GloVe vector file.
      (--glove_vocab|-gvoc) <vocab>  : GloVe vocabulary file with pairs
(word,frequency)
                                      and header.
      (--word2vec_vocab|-w2voc) <vocab> : Word2Vec vocabulary file with pairs
                                          (word,frequency).
      (--word_vector_input|-wvi)    : Input is text file and contains a trained
model.
      --file_index_dict|-fid        : Create a file index dict. Index entries are
words
                                      to lists of positions.
```

Παρόμοια με την περίπτωση του εργαλείου DataPrep, οι υποχρεωτικοί παράμετροι για την εκτέλεση του εργαλείου είναι ο καθορισμός εισόδου και εξόδου. Για τον καθορισμό εισόδου έχει προστεθεί η δυνατότητα επεξεργασίας .tbx (TermBase eXchange Files) και έχει αφαιρεθεί η δυνατότητα επεξεργασία μαζικών αρχείων από φακέλους και υποφακέλους. Όπως θα δούμε στην συνέχεια, έχουμε και την περίπτωση που το .xml αρχείο λειτουργεί επίσης και ως αναφορά στα πραγματικά δεδομένα που μπορεί να βρίσκονται σε πίνακες μίας βάσης δεδομένων. Αναλυτικά οι επιλογές που έχουμε είναι:

- `--morpho_dict` καθορισμός του δυαδικού μορφολογικού λεξικού που θα χρησιμοποιηθεί ως είσοδος για το εμπλουτισμό των ευρετηρίων με την κλίση ενός λήμματος.
- `--thes_dict` για τον καθορισμό του δυαδικού αρχείου θησαυρού που μπορεί να εμπλουτίσει τα ευρετήρια με συνώνυμες λέξεις.
- `--locale_dict` δείχνει στο λεξικό χαρακτήρων που χρησιμοποιούν τα δεδομένα που θα κτίσουν το λεξικό

- `--cust_lexicon` καθορίζει ένα λεξικό όρων σε .xml μορφή. Το σχήμα του .xml είναι ένα προκαθορισμένο σχήμα της Neurolingo. Μπορούμε να έχουμε πολλές εμφανίσεις της επιλογής αυτής. Τα λεξικά θα φορτωθούν ένα – ένα και θα ενοποιηθούν πριν ξεκινήσει η επεξεργασία τους.
- `--freqs_file` προσδιορίζει το αρχείο συχνοτήτων που μπορεί να παράγει η επεξεργασία των δεδομένων για την δημιουργία του λεξικού. Η κάθε γραμμή του αρχείου θα έχει μία λέξη και την συχνότητα της δηλ. τις εμφανίσεις της στους όρους του λεξικού. Συνδυάζεται με την επιλογή `--compute_freq` για τον καθορισμό της λειτουργίας υπολογισμού των συχνοτήτων.
- `--dump_qfile` δηλώνει το αρχείο κειμένου στο οποίο θα αποθηκευτούν τα q-grams που θα χρησιμοποιηθούν για το κτίσιμο του λεξικού. Συνδυάζεται υποχρεωτικά με την επιλογή `--dump_qgrams` που θα δούμε παρακάτω. Είναι επιλογές που μας επιτρέπουν να έχουμε καλύτερη εικόνα των δεδομένων που χρησιμοποιήθηκαν για το κτίσιμο του λεξικού.
- `--dump_rfile` χρησιμοποιείται σε συνδυασμό με την επιλογή `--dump_results` για να δηλώσουν ότι θέλουμε να εξάγουμε τα δεδομένα που θα χρησιμοποιηθούν για την κατασκευή του λεξικού σε ένα αρχείο .csv. Στην περίπτωση που τα δεδομένα είναι σε μία βάση δεδομένων μπορούμε να τα πάρουμε σε ένα .csv αρχείο κατά την διάρκεια της επεξεργασίας τους. Αν συνδυαστεί και με την επιλογή `--dump_only` τότε δηλώνουμε ότι θέλουμε μόνο την εξαγωγή των δεδομένων που θα χρησιμοποιηθούν στο κτίσιμο του λεξικού και όχι το λεξικό.
- `--dictionary` δηλώνει το αρχείο του λεξικού που θα δημιουργηθεί. Ο τύπος του λεξικού θα καθοριστεί από άλλες επιλογές.
- `–params` ορίζει ένα αριθμό από ζεύγη PARAM=VALUE που μπορούν να συμμετέχουν στην περίπτωση που τα δεδομένα είναι σε μία βάση δεδομένων. Με τον μηχανισμό αυτό μπορούμε στην γραμμή εκτέλεσης να καθορίσουμε στοιχεία σύνδεσης όπως username, password, κ.λπ.
- `--compute_freq` δείτε την περιγραφή για την επιλογή `--freqs_file`.
- `--dump_ids & --dump_freqs` σε συνδυασμό με την επιλογή `--compute_freq` καθορίζουν αν στο αρχείο συχνοτήτων που θα παράγουμε θα έχουμε μόνο την συχνότητα κάθε λέξης ή και το αναγνωριστικό (identifier) του όρου.
- `--word_index` χρησιμοποιείται στην περίπτωση που τα δεδομένα έρχονται από μία ερώτηση SQL, αποτελούνται από δύο ή περισσότερες λέξεις και θέλουμε να επιλέξουμε μία από αυτές για να συμμετέχει στην δημιουργία του λεξικού. Με την επιλογή αυτή καθορίζουμε την θέση της λέξης που θα χρησιμοποιηθεί στο λεξικό.
- `--create_text` προσομοιώνει την δημιουργία του δυαδικού λεξικού αλλά αντί για δυαδικό λεξικό δημιουργεί ένα αρχείο κειμένου με την πληροφορία που θα είχε το αντίστοιχο δυαδικό λεξικό. Χρήσιμη επιλογή για την εύρεση προβλημάτων σε σχέση με το περιεχόμενο που θα χρησιμοποιηθεί στην κατασκευή του λεξικού.
- `--db_index` δηλώνει ένα αρχείο .xml που έχει πληροφορία για την σύνδεση σε μία βάση δεδομένων και την ανάκτηση της πληροφορίας που χρειαζόμαστε για την κατασκευή του λεξικού. Όπως και στις υπόλοιπες περιπτώσεις, η επιλογή μπορεί να επαναληφθεί με συνέπεια να ενοποιήσουμε δεδομένα από δύο ή περισσότερες βάσεις.
- `--normalized` καθορίζει ότι όλες οι λέξεις θα κανονικοποιηθούν πριν ενταχθούν στην διαδικασία παραγωγής του λεξικού. Δηλ. οι χαρακτήρες θα μετατραπούν σε κεφαλαίους και άτονους.
- `--lemmatized` δηλώνει ότι για όλες οι λέξεις θα βρούμε το λήμμα στο οποίο ανήκουν και στην συνέχεια η κεφαλή του λήμματος θα αντικαταστήσει την λέξη στη επεξεργασία για την

δημιουργία του λεξικού. Για την εύρεση του λήμματος θα πρέπει να έχουμε δηλώσει και ένα μορφολογικό λεξικό. Δείτε την επιλογή `--morpho_dict`.

- `--tokenization_method` καθορίζει την μέθοδο που θα χρησιμοποιηθεί για την στοιχειοποίηση (tokenization) της πολυλεκτικής εισόδου. Η μέθοδος καθορίζεται με τους αριθμούς 0-3 και με ερμηνεία. 0 → διαχωρισμός στοιχείων με βάση τους κενούς χαρακτήρες, 1 → διαχωρισμός με βάση κενά και σύμβολα, 2 → χρήση διαχωριστών (π.χ. ‘,’) χωρίς παραμονή των διαχωριστών 3 → χρήση διαχωριστών και παραμονή τους ως στοιχεία. Οι κενοί χαρακτήρες και τα σύμβολα ορίζονται στο λεξικό χαρακτήρων όπως είδαμε στην επιλογή `--locale_dict`. Οι διαχωριστές μπορούν να ορισθούν στο .xml αρχείο που έχει τα στοιχεία σύνδεσης με την βάση όπως είδαμε στην επιλογή `--db_index`.
- `--normalization_method` καθορίζει την μέθοδο που θα χρησιμοποιηθεί για την κανονικοποίηση των λέξεων. Χρησιμοποιείται σε συνδυασμό με την επιλογή `--normalized` και οι τιμές και η ερμηνεία τους είναι: 0 → χωρίς κανονικοποίηση, 1→ απλή κανονικοποίηση με μετατροπή πεζών σε κεφαλαία και απομάκρυνση των τόνων, 2 → προχωρημένη κανονικοποίηση με ευρητικές μεθόδους για την περίπτωση λέξεων με μικτούς χαρακτήρες ελληνικούς και αγγλικούς. Ένα παράδειγμα χρήσης της μεθόδου 3 είναι η περίπτωση διαχείρισης ονομάτων που από λάθος πληκτρολόγησης το 1ο γράμμα τους είναι κεφαλαίο αγγλικό και τα υπόλοιπα ελληνικά. Θα μετατρέψει το αγγλικό γράμμα σε ελληνικό.
- `--use_only_one_lemma` δηλώνει ότι στην περίπτωση λημματοποίησης των λέξεων θα χρησιμοποιήσουμε μόνο το 1ο λήμμα αν υπάρχει ασάφεια και η λέξη ανήκει σε περισσότερα λήμματα. Συνδυάζεται με την επιλογή `--lemmatized`. Αν δεν καθοριστεί η επιλογή αυτή τότε όλα τα λήμματα θα ενσωματωθούν ως εναλλακτικές περιπτώσεις της λέξεις αυξάνωντας το μέγεθος του λεξικού.
- `--replace_with_lemma` δηλώνει ότι σε περίπτωση που βρούμε το λήμμα της λέξης, αυτό θα αντικαταστήσει την λέξη η οποία δεν θα εμφανιστεί στο λεξικό. Αν δεν βρεθεί λήμμα τότε θα χρησιμοποιηθεί η λέξη εισόδου. Αν δεν καθοριστεί η επιλογή αυτή τότε στην περίπτωση που έχουμε δηλώσει λημματοποίηση (`--lemmatized`) το λήμμα θα ενταχθεί ως εναλλακτική εκδοχή της εισόδιας λέξης. Δηλ. θα παραμείνουν και οι δύο λέξεις στο λεξικό.
- `--use_stem_on_unknown` καθορίζει ότι στην περίπτωση που θέλουμε λημματοποίηση και δεν βρεθεί το λήμμα θα προχωρήσουμε στην λειτουργία της απομάκρυνσης της κατάληξης (stemming) και το μέρος που θα μείνει θα χρησιμοποιηθεί ως λήμμα της λέξης.
- `--use_locale_norma` δηλώνει ότι για την κανονικοποίηση θα χρησιμοποιηθεί το λεξικό γραμμάτων.
- `--fill_morphology` δηλώνει ότι θα ενσωματωθούν στην επεξεργασία και τα μορφολογικά λήμματα των λέξεων. Η διαφορά με τις προηγούμενες μεθόδους είναι ότι η επιλογή αυτή ενεργεί στο τέλος της ανάγνωσης των δεδομένων και της δημιουργίας της ενδιάμεσης δομής στην μνήμη.
- `--expand` δηλώνει ότι οι λέξεις που θα χρησιμοποιηθούν στα ευρετήρια του λεξικού θα αναπτυχθούν με άλλες συνώνυμες λέξεις. Οι συνώνυμες λέξεις μπορούν να προέλθουν από πίνακες λέξεων που έχουν εμφανιστεί στα .xml αρχεία με στοιχεία για την πηγή (π.χ. βάση δεδομένων) ή εξωτερικά .xml αρχεία που ενσωματώνονται στην διαδικασία.
- `--expand_with_lemmas` δηλώνει τον εμπλουτισμό των ευρετηρίων με τα λήμματα των λέξεων. Χρησιμοποιεί το μορφολογικό λεξικό για την εύρεση του λήμματος.
- `--replace` δηλώνει την αντικατάσταση της λέξης ευρετηρίου με άλλη λέξη. Για τον ορισμό των εναλλακτικών λέξεων χρησιμοποιούμε πίνακες που ορίζονται μέσα στα .xml αρχεία.

- `--single` δηλώνει ότι έχουμε λεξικά μονολεκτικών όρων.
- `--combine` χρησιμοποιείται στην περίπτωση που υπάρχουν πολυλεκτικοί όροι όπου κάθε λέξη του όρου μπορεί να προέλθει από διαφορετική πολυλεκτική έκφραση. Για παράδειγμα έχουμε μία απάντηση από την βάση με τρεις στήλες. Η κάθε στήλη μπορεί να αποτελείται από μία ή περισσότερες λέξεις και ο συνδυασμός λέξεων που εμφανίζονται στις τρεις στήλες είναι όροι που πρέπει να μπουν στο λεξικό.
- `--concatenate` δηλώνει ότι το ευρετήριο θα πρέπει να εμπλουτιστεί με το περιεχόμενο του όρου σε περίπτωση που δεν έχει γίνει.
- `--use_term_stats` *για μελλοντική χρήση*
- `--dump_only` δείτε την περιγραφή στην επιλογή `--dump_rfile`
- `--dump_qgrams` δείτε την περιγραφή στην επιλογή `--dump_qfile`
- `--dump_results` δείτε την περιγραφή στην επιλογή `--dump_rfile`
- `--dump_sep` δηλώνει ότι θέλουμε στην έξοδο και τον διαχωριστή. Συνδυάζεται με την λειτουργικότητα που περιγράψαμε στην επιλογή `–dump_rfile`.
- `--qgrams` ορίζει τον αριθμό q για το σπάσιμο και επεξεργασία των q-grams
- `--auto_id` δηλώνει ότι για την περίπτωση που επεξεργαζόμαστε δεδομένα από .csv αρχεία δεν υπάρχει στήλη αναγνωριστικού της εγγραφής και θα πρέπει να χρησιμοποιήσουμε ένα αυτόματα παραγόμενο αναγνωριστικό.
- `--id_column` ορίζει την στήλη στο .csv αρχείο που περιέχει το αναγνωριστικό της εγγραφής.
- `--data_delimiter` ορίζει τον διαχωριστή που χρησιμοποιεί το .csv αρχείο με τα δεδομένα.
- `--data_columns` ορίζει τις στήλες του .csv αρχείου με τα δεδομένα προς επεξεργασία. Τα περιεχόμενα των στηλών θα ενωθούν πριν προχωρήσει η επεξεργασία στο ενοποιημένο κείμενο.
- `--num_columns` ορίζει τον αριθμό των στηλών που θα πρέπει να βρούμε στο .csv αρχείο. Λειτουργεί ως δικλείδα ασφαλείας για την περίπτωση λάθους .csv αρχείου.
- `--sample_count` περιορίζει τον αριθμό των εγγραφών που θα διαβάσουμε από την πηγή. Στην φάση της ανάπτυξης και ελέγχου μπορεί να θέλουμε να ελέγξουμε την επεξεργασία σε περιορισμένο αριθμό εγγραφών.
- `--no_sep` δηλώνει ότι η είσοδος είναι αρχεία κειμένου που περιέχουν μία λέξη ανά γραμμή.
- `--tab_sep` δηλώνει ότι στα αρχεία εισόδου .csv (ή .txt) ο διαχωριστής είναι ο χαρακτήρας tab (‘\t’)
- `--num_sep` δηλώνει ότι στα αρχεία εισόδου .txt (ή .csv) υπάρχουν δύο στήλες, η πρώτη με το αναγνωριστικό της εγγραφής και η δεύτερη με το κείμενο. Οι δύο στήλες χωρίζονται με κενό χαρακτήρα (ή tab ‘\t’)
- `--sep_expr` καθορίζει τον διαχωριστή σε αρχεία που έχουν πολλές τιμές ανά γραμμή (.csv ή .txt).
- `--skip_header_line` δηλώνει ότι η πρώτη γραμμή του αρχείου .csv ή .txt είναι επικεφαλίδα και δεν πρέπει να πάρει μέρος στην επεξεργασία.
- `--skip_double_entries` δηλώνει ότι κατά την επεξεργασία .csv αρχείων (ή .txt με εγγραφή ανά γραμμή) οι διπλές εγγραφές πρέπει να αγνοηθούν.
- `--use_disk_structs` δηλώνει ότι η επεξεργασία της συγκέντρωσης των δεδομένων του λεξικού και της επεξεργασίας τους θα χρησιμοποιήσουμε αρχεία στον δίσκο για τις ενδιάμεσες δομές. Η επιλογή αυτή είναι χρήσιμη όταν ο όγκος των δεδομένων που συγκεντρώνονται είναι μεγάλος και δεν επαρκεί η μνήμη του μηχανήματος για την διαχείρισή τους. Υπάρχει σημαντική

καθυστέρηση στην επεξεργασία με χρήση του δίσκου για τις ενδιάμεσες δομές και πρέπει να την χρησιμοποιούμε όταν δεν έχουμε άλλο τρόπο.

- `--use_memory_structs` δηλώνει ότι θα χρησιμοποιήσουμε την μνήμη για όλους προς χώρους αποθήκευσης στην κατασκευή του λεξικού. Είναι ο προκαθορισμένος τρόπος και δεν χρειάζεται να χρησιμοποιήσουμε αυτή την επιλογή. Υπάρχει ως ένδειξη για την εναλλακτική της επιλογής `–use_disk_structs`.
- `--tmp_folder` καθορίζει τον προσωρινό φάκελο που θα χρησιμοποιηθεί για να στεγάσει τα αρχεία που θα δημιουργηθούν με την επιλογή `–use_disk_structs`.
- `--bayes_dict` δηλώνει ότι το λεξικό που θα δημιουργηθεί θα είναι περιέχει πιθανότητες και μεγέθη που υπολόγισε η εκπαίδευση του αλγόριθμου Bayes.
- `--regression_dict` δηλώνει ότι το λεξικό που θα κτίσουμε θα περιέχει ένα μοντέλο Λογιστικής Παλινδρόμησης.
- `--compiled_dic` δηλώνει ότι το λεξικό που θα δημιουργηθεί θα είναι μια συμπυκνωμένη μορφή των δεδομένων εισόδου με κατάλληλα ευρετήρια για γρήγορη αξιοποίησή του. Χρησιμοποιείται σε συνδυασμό με το εργαλείο DataTrainer για την δημιουργία μοντέλων Bayes και Πολυώνυμης Λογιστικής Παλινδρόμησης.
- `--tfidf_dict` δηλώνει ότι θα κτίσουμε ένα λεξικό συχνοτήτων κατάλληλο να υπολογίσει αλγορίθμους όπως ο TF-IDF και BM25.
- `--inverted_index` καθορίζει ότι θα δημιουργηθεί ένα ανάστροφο λεξικό που θα έχει ευρετήριο από το αναγνωριστικό της εγγραφής προς λέξεις της εγγραφής. Για την περιοχή της Ηλεκτρονικής Λεξικογραφίας και της Επεξεργασίας Φυσικής Γλώσσας αυτό είναι το ανάστροφο λεξικό. Το κανονικό λεξικό είναι με ευρετήριο από την λέξη προς το αναγνωριστικό. Στη περίπτωση των εφαρμογών βάσεων δεδομένων η κατάσταση είναι αντεστραμμένη. Το κανονικό ευρετήριο είναι από τα αναγνωριστικά προς τις εγγραφές.
- `--parallel_train` δηλώνει ότι η εκπαίδευση του μοντέλου της Λογιστικής Παλινδρόμησης μπορεί να χρησιμοποιήσει πολλά νήματα και να γίνει παράλληλα αν το επιτρέπει το hardware.
- `--learning_rate` καθορίζει τον ρυθμό μάθησης στον ταξινομητή Λογιστικής Παλινδρόμησης.
- `--batch_size` καθορίζει τον αριθμό των τεμαχίων για την εκπαίδευση του ταξινομητή Λογιστικής Παλινδρόμησης.
- `--epoch_stuck_count` καθορίζει τον αριθμό των εποχών χωρίς πρόοδο κατά την εκπαίδευση του ταξινομητή Λογιστικής Παλινδρόμησης. Είναι μία από τις μεθόδους που μπορούμε να χρησιμοποιήσουμε για την παύση της εκπαίδευσης.
- `--missed_percentage` καθορίζει το ποσοστό των αποτυχημένων ταξινομήσεων κατά την φάση της εκπαίδευσης του ταξινομητή Λογιστικής Παλινδρόμησης. Και αυτή είναι μία συνθήκη εξόδου από την εκπαίδευση.
- `--print_train_info` δηλώνει ότι κατά την διάρκεια της εκπαίδευσης θα παραχθούν μηνύματα επεξήγησης της διαδικασίας.
- `--bigdata` δηλώνει την χρήση αρχείων του δίσκου για την δημιουργία του συμπυκνωμένου λεξικού που θα παραχθεί με την επιλογή `–compiled_dic`. Το ενδιάμεσο λεξικό Μηχανικής Μάθησης δημιουργείται από μεγάλες συλλογές και το μέγεθος του μπορεί να μην μπορεί να χωρέσει στην μνήμη του υπολογιστή.
- `--export_word_refs` δηλώνει ότι στο συμπυκνωμένο λεξικό που θα παραχθεί με την επιλογή `–compiled_dic` θα βάλουμε σε ξεχωριστή ενότητα και τις αναφορές κάθε λέξης, δηλ. σε ποιες εγγραφές/έγγραφα βρέθηκε η λέξη.

- `--word_vector` δηλώνει ότι θα δημιουργήσουμε ένα λεξικό με ενσωματώσεις λέξεων (word embeddings). Η είσοδος πρέπει να είναι αρχεία που βγήκαν από τα προγράμματα word2vec ή glove.
- `--word2vec` δηλώνει ότι η είσοδος είναι αρχεία του προγράμματος word2vec.
- `--glove` δηλώνει ότι η είσοδος είναι αρχεία του προγράμματος glove.
- `--glove_vocab` καθορίζει το αρχείο λεξιλογίου που παράγει το πρόγραμμα glove.
- `--word2vec_vocab` καθορίζει το αρχείο λεξιλογίου που παράγει το πρόγραμμα word2vec.
- `--word_vector_input` καθορίζει ότι η είσοδος θα είναι μοντέλα από ενσωματώσεις λέξεων. Με τις προηγούμενες επιλογές καθορίζουμε από ποιο πρόγραμμα δημιουργήθηκαν τα μοντέλα.
- `--file_index_dict` καθορίζει ότι θα δημιουργηθεί ένα ευρετήριο λέξεων προς σημεία εμφάνισής του στο αρχείο που δημιουργήθηκε από την ενοποίηση σώματος κειμένου. *Για μελλοντική χρήση*.

- Το εργαλείο `DataTrainer` δημιουργήθηκε για την εκπαίδευση μοντέλων **Bayes** και **Πολυώνυμης Λογιστικής Παλινδρόμησης** (**Multinomial Logistic Regression**). Δέχεται ως είσοδο ένα συμπυκνωμένο δυαδικό λεξικό που δημιουργήθηκε με το εργαλείο `DataCompile` με την επιλογή `--compiled_dic` όπως παρουσιάσαμε προηγουμένως.

```
Usage: DataTrainer <options>
   (--dictionary|-d) <data_dict.dic> : the compiled data dictionary created
                                       by DataCompiler
   (--model|-m) <model>  : model can be one of (bayes, multinomial)
   (--params|-p) <params> : the format of params is PARAM1=VAL1,PARAM2=VAL2,...
     bayes params
                  : binary=(true|false) create a binary NB model
   (--model_dict|-md) <data_dict.dic> : the dictionary that will store the result
                                        of training
   (--parallel|-pl)  : use of parallel training methods if possible.
```

  Στην περίπτωση του DataTrainer έχουμε λίγες επιλογές. Για τις παραμέτρους λειτουργίας που επηρεάζουν την διαδικασία της εκπαίδευσης των μοντέλων και αντικατοπτρίζουν τις **υπερπαραμέτρους** (**hyperparameters**) της Μηχανικής Μάθησης, έχουμε υιοθετήσει τον μηχανισμό ζευγών PARAM=VAL όπως θα δούμε. Αυτό μας επιτρέπει να προσθέτουμε παραμέτρους χωρίς να αλλάζουμε την διεπαφή με το εργαλείο. Οι επιλογές που υπάρχουν είναι:
  - `--dictionary` καθορίζει το συμπυκνωμένο δυαδικό λεξικό εισόδου που δημιουργήθηκε από τον DataCompiler.
  - `--model` καθορίζει τον τύπο του μοντέλου που θέλουμε να δημιουργήσουμε. Έχει δύο τιμές, την `bayes` για μοντέλο του ταξινομητή **Bayes** και την `multinomial` για το μοντέλο **Πολυώνυμης Μηχανικής Παλινδρόμησης**.
  - `–params` ορίζει τις υπερπαραμέτρους της εκπαίδευσης του μοντέλου. Ανάλογα με το μοντέλο που θέλουμε να εκπαιδεύσουμε έχουμε διαφορετικό σύνολο παραμέτρων. Για το μοντέλο Bayes υπάρχει μόνο μία παράμετρος Bool με το όνομα binary και καθορίζει το είδος του μοντέλου που θα δημιουργηθεί. Αν η τιμή είναι true τότε θα κτιστεί ένα δυαδικό μοντέλο ενώ σε διαφορετική περίπτωση πολυώνυμο. Για τα μοντέλα Πολυώνυμης Παλινδρόμησης οι παράμετροι είναι:
    - `batch_size` ορίζει το μέγεθος των τεμαχίων πάνω στα οποία θα γίνει η εκτίμηση του λάθους και της διόρθωσης των βαρών.

- `learning_rate` ορίζει τον ρυθμό μάθησης. Ο ρόλος της παραμέτρου αυτής στην εκπαίδευση του μοντέλου περιγράφεται στην ενότητα **Κάθοδος κλίσης (gradient descent function)**
- `learning_start` ορίζει την αρχική τιμή του ρυθμού μάθησης. Στην περίπτωση που ο ρυθμός μάθησης είναι μεταβλητός, μεγάλος στην αρχή και στην συνέχεια ελαττώνεται, η παράμετρος ρυθμίζει την αρχική τιμή.
- `missed_percentage` ορίζει το ανεκτό ποσοστό αποτυχημένης ταξινόμησης κατά τη διάρκεια της εκπαίδευσης.
- `epoch_stuck_count` ορίζει τον αριθμό των επαναλήψεων χωρίς πρόοδο που θα σταματήσει την εκπαίδευση.
- `use_bias` ορίζει την χρήση ή όχι του παράγοντα bias (πόλωση).
- `loss_function` ορίζει την συνάρτηση υπολογισμού της απώλειας. Δυνατές τιμές είναι:
  - `HingeBinary`
  - `HingeMultiLog`
  - `LogLoss`
  - `BinaryCross`
  - `CrossEntropy`
- `regularization` ορίζει τη μέθοδο κανονικοποίσης του αλγόριθμου. Οι επιτρεπτές τιμές είναι:
  - `No`
  - `L2`
  - `L1`
  - `ElasticNet`
- `tfidf_init`
- `verbose`
- `trace_key`

- `--model_dict` δηλώνει το δυαδικό λεξικό εξόδου που θα αποθηκεύσει το εκπαιδευμένο μοντέλο μηχανικής μάθησης.
- `--parallel` δηλώνει την χρήση παραλληλία στην εκπαίδευση του μοντέλου.

N-grams models - Αναγνώριση υποψήφιων όρων

Η εύρεση ορολογίας είναι μία από τις σημαντικές εργασίες του NLP. Οι όροι αποτελούνται κυρίως από ονοματικά σύνολα με ένα ή περισσότερα ουσιαστικά και επίθετα. Κατέχουν σημαντικό ρόλο τόσο στην κατανόηση του περιεχομένου όσο και στην σωστή ευρετηρίασή του. Τα μοντέλα αυτά έχουν ως στόχο την δημιουργία στατιστικών κατανομών που να εκφράζουν την πιθανότητα εμφάνισης μιας ακολουθίας λέξεων. Προσπαθούν να απαντήσουν στο ερώτημα «αν έχουμε τις λέξεις w_1, w_2, …, w_{n-1}» ποια είναι η πιθανότητα να ακολουθεί η λέξη "w_n". Η απάντηση απαιτεί τον υπολογισμό των πιθανοτήτων όλων των συνδυασμών μεγάλων ακολουθιών λέξεων που είναι δύσκολο να πραγματοποιηθεί ακόμα και σε υψηλής απόδοσης υπολογιστές (CPU & Memory). Για να απλοποιηθεί το πρόβλημα, υιοθετούμε την παραδοχή του Markov, που προτείνει ότι οι λέξεις έχουν μεγάλη εξάρτηση με τις γειτονικές τους λέξεις και όσο απομακρυνόμαστε από αυτές τόσο μειώνεται η εξάρτηση. Η παραδοχή αυτή μας επιτρέπει να μειώσουμε σημαντικά το χώρο εξέτασης και υπολογισμού πιθανοτήτων σε μικρές αποστάσεις, συνήθως 2-5. Για παράδειγμα αν θεωρήσουμε ότι οι λέξεις επηρεάζονται μόνο από αυτές που προηγούνται, έχουμε ένα

μοντέλο ζευγών όπου για κάθε λέξη υπολογίζουμε την πιθανότητα εμφάνισής της μετά την ακολουθία «w_1, w_2, …, w_{n-1}» από τον παρακάτω τύπο

$$P(w_n|w_1,w_2,w_3,\ldots,w_{n-1}) = P(w_2|w_1)\ P(w_3|w_2)\ \ldots\ P(w_{n-1}|w_{n-2})\ P(w_n|w_{n-1}) \quad (1)$$

Το μοντέλο αυτό ονομάζεται 2-gram και με παρόμοιο τρόπο ορίζονται τα μοντέλα 3-grams, 4-grams, κ.λπ. Ο τρόπος υπολογισμού των πιθανοτήτων $P(w_i|w_{i-1})$ ονομάζεται «Υπολογισμός της Μέγιστης Ομοιότητας» (Maximum Likelihood Estimation – MLE) που υπολογίζει την πιθανότητα μετρώντας τις συχνότητες εμφάνισης όλων των ζευγών $C(w_{n-1},w_n)$ σε ένα σώμα κειμένων. Ο τύπος υπολογισμού των συχνοτήτων για την περίπτωση του μοντέλου 2-gram είναι:

$$P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)}{\sum_w C(w_{n-1}w)} \quad (2)$$

Παρόμοια ορίζεται και η γενική περίπτωση μοντέλου n-gram. Τα μοντέλα που έχουν μεγαλύτερη ακρίβεια και χρησιμότητα είναι τα 3-grams, 4-grams και 5-grams αν υπάρχει κατάλληλο σε ποσότητα και ποιότητα σώμα για τον υπολογισμό τους.

Τα μοντέλα n-grams είναι χρήσιμα στην εύρεση πολυλεκτικών ονοματικών συνόλων και όπως θα δούμε παρακάτω στην παραγωγή κειμένου. Για την εύρεση πολυλεκτικών συνόλων, π.χ. ονόματα ανθρώπων, οργανισμών και όρων μπορούμε να αυξήσουμε την ποιότητα του αποτελέσματος αν συνδυάσουμε το αποτέλεσμα με ταίριασμα μορφοσυντακτικών χαρακτηριστικών των n-λέξεων. Αυτή τη μέθοδο εφαρμόσαμε στην ανάλυση των κειμένων του «Φωτόδεντρο» με στόχο την ανακάλυψη υποψήφιων όρων, δηλ. ακολουθίες λέξεων που εκφράζουν μία έννοια ή μία οντότητα. Χρησιμοποιώντας τον φορμαλισμό ΚΑΝΟΝ ορίσαμε τα μορφοσυντακτικά σχεδιότυπα που ακολουθούν οι όροι και σε συνδυασμό με τον υπολογισμό των συχνοτήτων των μοντέλων n-grams δημιουργήθηκε ένα μοντέλο με βελτιωμένη ακρίβεια στον υπολογισμό των υποψήφιων όρων. Στο Παράρτημα Α έχουμε τους μορφοσυντακτικούς κανόνες που χρησιμοποιήσαμε. Ένα παράδειγμα κανόνα παρουσιάζεται παρακάτω:

```
/* TERMS_1_4 */
[ARULE="TERMS_1_4", VTEXT="TERM", LEXY=GNC_Reduction(1,[N],[N],"%1 %3"),
                                  TTEXT=Concatenate("TTEXT"," ")] =>
  \
    [LEXY->HasMAttrsAndNotMA([N],[PRON,V,ART,INTERJ])],
    [LEXY->HasMAttrs([PREP])],
    [LEXY->HasMAttrsAndNotMA([N],[PRON,V,ART,INTERJ])]
  / ;
```

Ο κανόνας αναγνωρίζει 3-grams τα οποία αποτελούνται από τριάδες (Ουσιαστικό/N, Πρόθεση/PREP, Ουσιαστικό/N). Παραδείγματα από τριάδες που αναγνώρισε ο κανόνας στα κείμενα του «Φωτόδεντρο» είναι:

```
περιβάλλον/5 με/1 αέρια/1
όργωμα/1 με/1 βόδια/1
ατμόσφαιρα/2 με/2 καυσαέρια/1
απόσταση/1 σε/1 χιλιόμετρα/1
```

Μια εφαρμογή των γλωσσικών μοντέλων n-grams είναι η παραγωγή κειμένου. Η διαδικασία παραγωγής είναι απλή και στηρίζεται στην δημιουργία και ευρετηρίαση των n-grams σε πίνακα. Ξεκινώντας από λέξεις που έχουν εμφανιστεί ως αρχή προτάσεων και επιλέγοντας τυχαία κάθε φορά την επόμενη με βάση τις προηγούμενες n-1 λέξεις παράγουμε προτάσεις. Για την διευκόλυνση της διαδικασίας εισάγουμε 2 απλές τεχνητές λέξεις, την <s> που δηλώνει την αρχή της πρότασης και την </s> που δηλώνει το τέλος. Όταν μια λέξη ξεκινά μια πρόταση, το n-gram που θα δημιουργηθεί έχει n-1 <s> και την 1η λέξη της πρότασης. Στην συνέχεια κάθε λέξη που εμφανίζεται δημιουργεί ένα νέο n-gram με τις n-1 τελευταίες λέξεις του

προηγούμενου n-gram και με τελευταία την εισόδια λέξη. Παρακάτω παραθέτουμε κάποιες προτάσεις που δημιουργήθηκαν από τα n-grams (3-grams, 4-grams, 5-grams) από την επεξεργασία των βιβλίων Γεωγραφίας.

```
<<<
Επειδή η ρωσική πεδιάδα είναι αρκετά μεγάλη μόνο σ αυτήν υπάρχουν μεγάλα σε μήκος
ποτάμια με εξαίρεση τον Δούναβη που ξεκινά από το Σικάγο και καταλήγει στο
Πίτσμπουργκ
Υπολογίζεται πως κύματα ύψους δεκάδων μέτρων και ταχύτητας χιλιομέτρων την ώρα
σάρωσαν λίγο μετά την έκρηξη τα βόρεια παράλια της Κρήτης και τις ακτές του
Ατλαντικού από την έρημο Σαχάρα το κεντρικό τμήμα της που έχει πολλά τροπικά δάση και
σαβάνες και το νότιο τμήμα της που πλησιάζει τον Ισημερινό αποτελεί τη γέφυρα μεταξύ
Βόρειας και Νότιας Αμερικής
Κάτι που δεν ήξεραν οι Αιγύπτιοι ήταν ότι οι πληθυσμοί των ελαφιών και των καριμπού
των ταράνδων αυξήθηκαν καταστρέφοντας τη βλάστηση
Χαρακτηριστικός βόρειος άνεμος στην κεντρική Μακεδονία τον χειμώνα είναι ο βαρδάρης
ενώ στο τέλος της διαδρομής του επιτρέπουν την παραγωγή ηλεκτρικής ενέργειας
βασίζεται σε μεγάλο ποσοστό είναι σκαμμένη μέσα σε βράχια και εξυπηρετεί τη διέλευση
πλοίων κάθε είδους ακόμη και των γιγαντιαίων σύγχρονων πετρελαιοφόρων
Δυστυχώς όμως δεν έχουν κατασκευαστεί ακόμη συσκευές που θα μας έκανε να περιμένουμε
μια ανάλογα σημαντική ποικιλία φυσικής βλάστησης
Ας εντοπίσουμε και άλλους παράγοντες που επηρέασαν τη γεωγραφική θέση του Μεξικού και
εκβάλλει στον Ινδικό Ωκεανό
Απλώνεται στις βόρειες περιοχές της Θράκης και οι πιο γνωστές κορυφές της είναι το
Ελσίνκι
>>>
```

Ταξινομητής Bayes

Η ταξινόμηση είναι μία από τις σημαντικότερες λειτουργίες του υπολογιστή ανεξάρτητα από το είδος της εφαρμογής που τρέχει. Η πράξη της ταξινόμησης θεωρείται επίσης και ως το κέντρο της ανθρώπινης νοημοσύνης μιας και προηγείται κάθε απόφασης μας. Με την ραγδαία εξέλιξη των τεχνολογιών Τεχνητής Νοημοσύνης τα τελευταία χρόνια έχουμε μεγάλη αύξηση στην εφαρμογή στοχαστικών αλγορίθμων ταξινόμησης. Οι αλγόριθμοι αυτοί χωρίζονται σε δύο κατηγορίες, τους εποπτευόμενους (supervised) αλγορίθμους και τους μη-εποπτευόμενους (unsupervised). Η κατηγορία των εποπτευόμενων αλγορίθμων στηρίζεται στο ότι υπάρχουν δεδομένα που έχουν ήδη σηματοδοτηθεί από τον άνθρωπο και στόχος των αλγορίθμων είναι να «καταλάβουν» τον τρόπο της σηματοδότησης και να μπορέσουν να τον εφαρμόσουν σε άγνωστα δεδομένα. Ένας από τους γνωστότερους και βασικούς στοχαστικούς αλγόριθμους που έχουν χρησιμοποιηθεί για την ταξινόμηση (classification) των δεδομένων βασίζεται στο θεώρημα του Bayes. Ο αλγόριθμος χρησιμοποιήθηκε στην **κατηγοριοποίηση κειμένων** (text categorization) αλλά και στην **ανάλυση συναισθήματος** (sentiment analysis). Η κατηγοριοποίηση ενός κειμένου έχει στόχο να κατατάξει ένα κείμενο σε μία κατηγορία. Για παράδειγμα ένα δημοσιογραφικό άρθρο στις πιθανές κατηγορίες που έχει το μέσο ενημέρωσης όπως: αθλητική, πολιτική, πολιτιστική, κ.α. Ως βάση εκπαίδευσης μπορεί να χρησιμοποιηθεί το ιστορικό αρχείο του μέσου όπου υπάρχουν τα παλαιότερα άρθρα χαρακτηρισμένα. Άλλη περίπτωση που χρησιμοποιήθηκε με επιτυχία το θεώρημα Bayes αφορά την ανίχνευση κακόβουλης ή ανεπιθύμητης ηλεκτρονικής αλληλογραφίας. Η εφαρμογή εκπαιδεύεται από τον χαρακτηρισμό που κάνει ο χρήστης στην εισερχόμενη αλληλογραφία και βελτιώνει την απόδοσή της με τον χρόνο.

Στην ανάλυση συναισθήματος έχουμε εφαρμογές βαθμολόγησης με θετικό ή αρνητικό πρόσημο αντικείμενα ενδιαφέροντος μεγάλης ομάδας ανθρώπων. Για παράδειγμα, βιβλία, κινηματογραφικές ταινίες, τουριστικά αξιοθέατα, αθλητικοί αγώνες, κ.λπ. ενδιαφέρουν μεγάλες ομάδες ανθρώπων που

εκφράζουν την άποψή τους σε ηλεκτρονικά μέσα. Σε κάποιες περιπτώσεις, π.χ. tripadvisor (https://www.tripadvisor.com.gr/) η ανθρώπινη βαθμολόγηση μπορεί να έχει διαβαθμίσεις θετικού ή αρνητικού πρόσημου.

Το θεώρημα του Bayes έχει χρησιμοποιηθεί στην δημιουργία μιας κατηγορίας ταξινομητών με την ονομασία Naïve Bayes Classifiers (Απλοϊκοί ταξινομητές Bayes). Το «απλοϊκοί» χαρακτηρίζει την παραδοχή των ταξινομητών να θεωρούν ότι τα χαρακτηριστικά τα οποία παίρνουμε υπόψη στην εκπαίδευση του μοντέλου είναι ανεξάρτητα μεταξύ τους κάτι που γενικά δεν ισχύει. Παρόλα αυτά οι ταξινομητές αυτοί έχουν δώσει πολύ καλά αποτελέσματα και χρησιμοποιούνται με μεγάλη επιτυχία σε πολλές εφαρμογές εποπτευόμενης μηχανικής μάθησης.

Αναγκαία προϋπόθεση για την εφαρμογή των ταξινομητών Bayes είναι η εύρεση των χαρακτηριστικών που θα χρησιμοποιηθούν στην κατηγοριοποίηση. Η εύρεση των κατάλληλων χαρακτηριστικών είναι ένα από τα μεγάλα προβλήματα του NLP. Μία από τις αρχικές απλουστεύσεις που κάνουμε είναι να θεωρήσουμε ότι ένα έγγραφο είναι ένα «**σακούλι με λέξεις**» (**bag of words**), δηλ. έχουμε αυτόνομες και ανεξάρτητες λέξεις που εκφράζουν το νόημα του κειμένου, η σειρά τους δεν παίζει ρόλο. Τα χαρακτηριστικά που θα επιλέξουμε εξαρτώνται από τις λέξεις που έχουμε στο σακούλι και κυρίως τις κατηγορίες που έχουμε επιλέξει. Για παράδειγμα αν έχουμε επιλέξει τα ουσιαστικά ως ένα χαρακτηριστικό, ο αριθμός των ουσιαστικών στο σακούλι θα μας δώσει το μέγεθος του χαρακτηριστικού αυτού. Άλλα χαρακτηριστικά που μπορεί να έχουμε είναι, λέξεις που εκφράζουν θετικό συναίσθημα και οι οποίες θα τις βρούμε σε ένα λεξικό θετικών όρων, λέξεις που εκφράζουν αρνητικό συναίσθημα, λέξεις που εκφράζουν μέγεθος όπως επίθετα σε συγκριτικό ή υπερθετικό βαθμό, κ.λπ. Αφού επιλέξουμε τα χαρακτηριστικά που θα χρησιμοποιήσουμε για τον ταξινομητή μας, έρχεται η ώρα να εκπαιδεύσουμε το μοντέλο μας από ένα σύνολο κειμένων όπου για κάθε κείμενο έχουμε υπολογίσει τα χαρακτηριστικά αυτά και ξέρουμε την κατηγορία του. Η εκπαίδευση του μοντέλου αφορά τον υπολογισμό του βάρους που έχει κάθε χαρακτηριστικό για την κάθε κατηγορία έτσι ώστε ο συνδυασμός των βαρών να δώσει στην κατηγορία του κειμένου το μεγαλύτερο βάρος. Για να απλοποιήσουμε ακόμα παραπάνω την διαδικασία και επειδή η εύρεση των χαρακτηριστικών και ο υπολογισμός τους από τον άνθρωπο είναι μια δαπανηρή διαδικασία, θεωρούμε ότι κάθε λέξη (ή συγκεκριμένη κατηγορία λέξεων όπως ουσιαστικά και επίθετα) είναι ένα ξεχωριστό χαρακτηριστικό. Στην περίπτωση αυτή η μέτρηση ενός χαρακτηριστικού ανάγεται στην μέτρηση του αριθμού εμφάνισης μιας λέξης σε ένα κείμενο. Η παρακάτω φόρμουλα περιγράφει τον υπολογισμό της πιθανότητας ενός εγγράφου $\mathtt{d}$ το οποίο περιέχει τις λέξεις $\mathtt{w}_1$, $\mathtt{w}_2$, …, $\mathtt{w}_n$ να ανήκει στη κατηγορία $\mathtt{c}$.

$$P(d|c) \cdot P(c) = P(w_1 w_2 \dots w_n|c) \cdot P(c) = P(w_1|c) \cdot P(w_2|c) \cdot \dots \cdot P(w_n|c) \cdot P(c) \quad (3)$$

Στην παραπάνω φόρμουλα το $P(c)$ εκφράζει την πιθανότητα να έχουμε την κατηγορία c, το $P(d|c)$ την πιθανότητα το έγγραφο d να ανήκει στην κατηγορία c και το $P(w_i|c)$ την πιθανότητα η λέξη w_i να ανήκει στην κατηγορία c. Η εύρεση της σωστής κατηγορίας ανάγεται στην εύρεση της μεγαλύτερης τιμής για τις πιθανότητες όλων των κατηγοριών και αποτυπώνεται στην παρακάτω φόρμουλα.

$$c_B = \underbrace{argmax}_{c \in C} P(c) \cdot \prod_{w \in V} P(w|c) \quad (4)$$

Το c_B εκφράζει την κατηγορία που επιλέγει ο ταξινομητής Bayes, το V το λεξιλόγιο των εγγράφων, το C το σύνολο όλων των κατηγοριών. Η συνάρτηση argmax εκφράζει τον υπολογισμό της μεγαλύτερης τιμής, εδώ πιθανότητας από τις τιμές που υπολογίζει το εσωτερικό γινόμενο.

Για την εκπαίδευση του μοντέλου χρειαζόμαστε ένα σύνολο από έγγραφα με σημαδεμένη την κατηγορία τους. Η εκπαίδευση του μοντέλου συμπεριλαμβάνει την διάσπαση του κάθε κειμένου σε λέξεις, την μέτρηση της συχνότητες κάθε λέξης και τον υπολογισμό του βάρους της κάθε λέξης για κάθε μία από τις

κατηγορίες που έχουμε. Ο αριθμός των εγγράφων ανά κατηγορία θα επιτρέψει τον υπολογισμό του $P(c)$, της πιθανότητας δηλ. να έχουμε την κατηγορία c. Τέλος το μοντέλο με τα βάρη των λέξεων-χαρακτηριστικών καθώς και τα βάρη τους ανά κατηγορία θα αποθηκευτεί σε ένα ειδικό λεξικό που θα ενσωματώνει επίσης ένα ευρετήριο λέξεων για γρήγορη προσπέλαση της πληροφορίας. Οι παρακάτω φόρμουλες μας δίνουν τον τρόπο υπολογισμού των πιθανοτήτων που χρειάζεται ο ταξινομητής Bayes. Πρώτα έχουμε την φόρμουλα που υπολογίζει την πιθανότητα κάθε κατηγορίας, δηλ. $P(c)$. Το καπέλο στην φόρμουλα εκφράζει ότι έχουμε εκτίμηση της τιμής και όχι την απόλυτη τιμή η οποία είναι άγνωστη.

$$\hat{P}(c) = \frac{N_c}{N_{doc}} \quad (5)$$

Το N_c είναι ο αριθμός των εγγράφων που έχουν μαρκαριστεί ότι ανήκουν στην κατηγορία c και το N_{doc} ο συνολικός αριθμός των εγγράφων στη συλλογή που χρησιμοποιήσαμε για την εκπαίδευση του μοντέλου. Αντίστοιχα για την εκτίμηση των $P(w_i|c)$ έχουμε την παρακάτω φόρμουλα.

$$\hat{P}(w_i|c) = \frac{count(w_i,c)}{\sum_{w \in V} count(w,c)} \quad (6)$$

Όπου V είναι το λεξιλόγιο της συλλογής, δηλ. η ένωση όλων των λέξεων των εγγράφων.

Το λεξικό που αποθηκεύει το μοντέλο του Bayes θα χρησιμοποιηθεί από τον ταξινομητή Bayes ο οποίος για κάθε λέξη του εγγράφου εισόδου για το οποίο θέλουμε να βρούμε την κατηγορία του και για κάθε κατηγορία που υπάρχει υπολογίζει την πιθανότητα από την φόρμουλα (4).

Ταξινομητής Λογιστικής Παλινδρόμησης (Logistic Regression Classifier)

Η ταξινόμηση της λογιστικής παλινδρόμησης (logistic regression classification) αποτελεί ένα από σημαντικότερα εργαλεία ανάλυσης δεδομένων σε πολλούς επιστημονικούς κλάδους. Ακόμα και τα νευρωνικά δίκτυα που είναι η βάση της μηχανικής μάθησης και κυρίως της βαθιάς μηχανικής μάθησης, είναι ακολουθίες από ταξινομητές λογιστικής παλινδρόμησης.

Η απλή χρήση του ταξινομητή χρησιμοποιείται για την ταξινόμηση των δεδομένων ανάλογα με την εμφάνιση ενός συνόλου χαρακτηριστικών. Είναι η περίπτωση του δυαδικού ταξινομητή επειδή ταξινομεί τα δεδομένα σε δύο κατηγορίες που μπορεί να τις αντιστοιχίσουμε στις τιμές 1 (ή θετική κατηγορία) και 0 (αρνητική κατηγορία). Το επόμενο επίπεδο ταξινόμησης είναι ο πολυώνυμος ταξινομητής (multinomial classifier) με δυνατότητα να ταξινομήσει τα δεδομένα σε ένα σύνολο από περισσότερες από δύο κατηγορίες. Μία παραλλαγή του πολυώνυμου ταξινομητή είναι η ταξινόμηση των δεδομένων σε δύο ή περισσότερες κατηγορίες.

Παρόμοια με τον ταξινομητή Bayes, ανήκει στην κατηγορία των εποπτευόμενων ταξινομητών οι οποίοι εκπαιδεύουν ένα μοντέλο με δεδομένα που έχουν μαρκαριστεί από τον άνθρωπο με τιμές για ένα σύνολο χαρακτηριστικών. Στην συνέχεια το μοντέλο αυτό χρησιμοποιείται σε άγνωστα δεδομένα για την εύρεση των τιμών των επίμαχων χαρακτηριστικών.

Η εκπαίδευση του ταξινομητή στηρίζεται στην ύπαρξη χαρακτηρισμένων δεδομένων ικανού όγκου. Αν έχουμε μαρκάρει M κείμενα με την ύπαρξη n χαρακτηριστικών (features). Ο τρόπος που χρησιμοποιούμε για την αναπαράσταση των χαρακτηριστικών είναι ένα άνυσμα n θέσεων, π.χ. $[x_1, x_2, \dots, x_n]$. Για τα κείμενα χρησιμοποιούμε την αναπαράσταση $D^{(j)}$ όπου ο εκθέτης εκφράζει το j κείμενο. Όταν θέλουμε να αναφερθούμε στο χαρακτηριστικό i του κειμένου j τότε γράφουμε $D_i^{(j)}$. Ο ταξινομητής αποτελείται από τρία συστατικά.

1. Την συνάρτηση της ταξινόμησης που υπολογίζει την τιμή $\hat{y}$ που είναι η εκτίμηση της πραγματικής τιμής $p(y|x)$ όπως είδαμε και στον ταξινομητή Bayes. Ως συνάρτηση χρησιμοποιείται η σιγμοειδής[4] (**sigmoid**) συνάρτηση σε συνδυασμό με την **softmax**[5] (**μεγίστη**) όπως θα δούμε στην συνέχεια.
2. Μία συνάρτηση για την στοχευμένη εκπαίδευση η οποία συνήθως προσπαθεί να ελαχιστοποιήσει το λάθος στο υπολογισμό κατά την διάρκεια της εκπαίδευσης. Στην δική μας περίπτωση θα χρησιμοποιήσουμε την συνάρτηση που είναι γνωστή με **απώλεια διασταυρούμενης εντροπίας** (**cross-entropy loss function**).
3. Μία συνάρτηση βελτιστοποίησης της συνάρτησης του ταξινομητή με μείωση του λάθους που κάνει. Θα δούμε την συνάρτηση **στοχαστικής καθόδου κλίσης** (**stochastic gradient descent**).

Ο αλγόριθμος λειτουργεί σε δύο φάσεις.

- **Εκπαίδευση** (**training**): κατά την διάρκεια της εκπαίδευσης υπολογίζονται τα βάρη w και b που θα δούμε στην συνέχεια.
- **Έλεγχος** (**test**): Δεδομένου μίας τιμής ελέγχου x υπολογίζουμε την τιμή $p(y|x)$ που είναι η μεγαλύτερη πιθανότητα για τις τιμές $y = 0$ ή $y = 1$

Η σιγμοειδής (sigmoid) συνάρτηση

Η συνάρτηση πήρε το όνομά της λόγω της αναπαράστασης της που μοιάζει με τελικό σίγμα. Ο υπολογισμός της τιμής πρέπει να καθορίσει την κλάση που θα ταξινομήσει το κείμενο $D^{(j)}$ σε μία από τις κατηγορίες 1 ή 0. Η τιμή όπως θα δούμε είναι ένας πραγματικός αριθμός στην κλίμακα $0 \ldots 1$. Για να την απεικονίσουμε σε μία από τις δύο τιμές 1 και 0 επιλέγουμε τις τιμές $0 \ldots \frac{1}{2}$ στην τιμή 0 και για τιμές $\frac{1}{2} \ldots 1$ στην τιμή 1. Στον υπολογισμό της τιμής χρησιμοποιούμε το άνυσμα των χαρακτηριστικών $[x_1, x_2, \ldots, x_n]$ του κειμένου, ένα άνυσμα βαρών $[w_1, w_2, \ldots, w_n]$ που αντιστοιχεί ένα βάρος w_k στο χαρακτηριστικό w_k και ένα παράγοντα κλίσης (bias) b. Η φόρμουλα είναι:

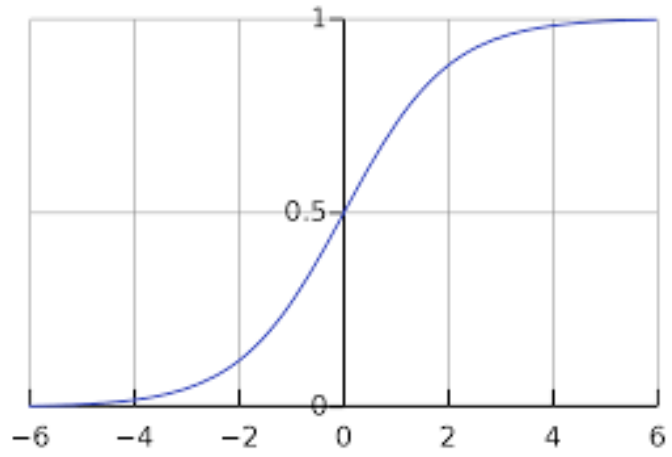
$$z = \left(\sum_{i=1}^{n} w_i x_i\right) + b \tag{7}$$

Η φόρμουλα υπολογίζει το εσωτερικό γινόμενο δύο ανυσμάτων $w \cdot x$ και στην τελική τιμή προσθέτουμε τον παράγοντα b. Η κύρια δουλειά του αλγόριθμου της λογιστικής παλινδρόμησης είναι να υπολογίσει τις τιμές των βαρών w_i και του b. Οι τιμές της z είναι στην κλίμακα $-\infty \ldots +\infty$. Για να μπορέσει να χρησιμοποιηθεί ως πιθανότητα και να καθορίσει μία από τις κλάσεις 1 ή 0, περνάμε την τιμή από την σιγμοειδή συνάρτηση $\sigma(z)$ η οποία παρουσιάζεται στην φόρμουλα (8) παρακάτω.

$$y = \sigma(z) = \sigma(w \cdot x + b) = \frac{1}{1 + e^{-z}} \tag{8}$$

Στο παρακάτω διάγραμμα βλέπουμε τις τιμές που μπορεί να πάρει η τιμή της μεταβλητής y καθώς και την τιμή $\frac{1}{2} = 0.5$ η οποία διαχωρίζει τις δύο κλάσεις.

[4] https://el.wikipedia.org/wiki/Σιγμοειδής_συνάρτηση
[5]



Το άνυσμα των χαρακτηριστικών το οποίο χρησιμοποιείται για τον υπολογισμό της κλάσης εξαρτάται από την συγκεκριμένη εφαρμογή. Η σωστή επιλογή τους είναι σημαντικός παράγοντας για την απόδοση του αλγόριθμου. Κάποια παραδείγματα για τις εφαρμογές επεξεργασίας κειμένων μπορεί να είναι:

- Αριθμός λέξεων
- Αριθμός ουσιαστικών
- Αριθμός λέξεων που βρίσκονται σε κάποιο λεξικό
- Αριθμός του χαρακτήρα '!'
- …

Εκπαίδευση και συνάρτηση cross-entropy loss

Η εκπαίδευση του ταξινομητή λογιστικής παλινδρόμησης επικεντρώνεται στον υπολογισμό των βαρών w_i και του παράγοντα b. Για τον υπολογισμό των τιμών χρειαζόμαστε ένα σύνολο από προϋπολογισμένα δεδομένα και αρχικές τιμές για τα βάρη και τον παράγοντα κλίσης. Η εκπαίδευση γίνεται σε επαναλαμβανόμενες φάσεις όπου περιλαμβάνουν

1. Υπολογισμό της τιμής $\hat{y}$ που είναι η εκτίμηση της τελικής τιμής.
2. Σύγκριση της τιμής $\hat{y}$ με την προϋπολογισμένη τιμή y και υπολογισμός του λάθους.
3. Διόρθωση της τιμής $\hat{y}$ με τροποποιήσεις των βαρών και της κλίσης.

Το 1ο βήμα το είδαμε στην προηγούμενη παράγραφο με τον υπολογισμό της σιγμοειδούς συνάρτηση. Στο 2ο βήμα αφορά τον υπολογισμό του λάθους που γίνεται με εφαρμογή της συνάρτησης cross-entropy loss που είναι μία συνάρτηση κόστους. Στην επόμενη παράγραφο θα δούμε τον τρόπο που διορθώνουμε τις τιμές των βαρών και του παράγοντα κλίσης με την εφαρμογής της συνάρτησης **στοχαστική κάθοδος κλίσης** (stochastic gradient descent). Η συνάρτηση απώλειας (loss function) έχει την μορφή $L(\hat{y}, y)$ και εκφράζει το πόσο διαφέρει η υπολογισμένη από τον ταξινομητή τιμή $\hat{y}$ από την πραγματική προϋπολογισμένη τιμή y.

Στην περίπτωση του δυαδικού ταξινομητή όπου οι έγκυρες τιμές είναι 0 και 1. Ψάχνουμε να βρούμε την καλύτερη εκτίμηση που μπορούμε να κάνουμε με συνθήκη τις τιμές του ανύσματος των βαρών και του παράγοντα της κλίσης. Η επιλογή των τιμών για τα βάρη και την κλίση πρέπει να μεγιστοποιεί την πιθανότητα να παράγουμε ταμπέλες παρόμοιες με αυτές που έχουν προϋπολογιστεί για τα δεδομένα που χρησιμοποιούμε στην εκπαίδευση. Επειδή οι υπολογισμοί που γίνονται παράγουν πολύ μικρές ή πολύ μεγάλες τιμές και υπάρχει κίνδυνος υπερχείλισης, μεταφέρουμε την επεξεργασία στον λογαριθμικό χώρο. Για τον υπολογισμό της πιθανότητας χρησιμοποιούμε τον τύπο υπολογισμού πιθανότητα από την κατανομή Bernoulli.

$$p(y|x) = \hat{y}^y(1-\hat{y})^{1-y} \quad (9)$$

Μεταφέροντας την παραπάνω εξίσωση στον λογαριθμικό χώρο και απλοποιώντας καταλήγουμε στην παρακάτω εξίσωση.

$$\log p(y|x) = y \log \hat{y} + (1-y) \log(1-\hat{y}) \quad (10)$$

Η παραπάνω εξίσωση εκφράζει την πιθανότητα που πρέπει να μεγιστοποιήσουμε. Για να πάρουμε την συνάρτηση απώλειας, δηλ. πόσο απέχουμε από την σωστή τιμή, αντιστρέφουμε το πρόσημο. Η τιμή που θα πάρουμε είναι η **απώλεια διασταυρούμενης απώλειας** (**cross-entropy loss**) και δίνεται από την εξίσωση.

$$L_{CE}(\hat{y}, y) = -\log p(y|x) = -[y \log \hat{y} + (1-y) \log(1-\hat{y})]$$

Και αντικαθιστώντας το $\hat{y}$ με την έκφραση $\sigma(w \cdot x + b)$ παίρνουμε την τελική εξίσωση υπολογισμού της απώλειας διασταυρωμένης εντροπίας.

$$L_{CE}(\hat{y}, y) = -[y \log \sigma(w \cdot x + b) + (1-y) \log\big(1 - \sigma(\widehat{w \cdot x + b})\big)] \quad (11)$$

Κάθοδος κλίσης (gradient descent function)

Το τρίτο και τελευταίο βήμα αφορά τις μεταβολές στα βάρη και στον παράγοντα κλίσης ώστε ο υπολογισμός της κλάσης να διορθώσει το λάθος και να μειώσει την απώλεια που υπολογίσαμε στο προηγούμενο βήμα. Για να βρούμε τις καλύτερες τιμές που ελαχιστοποιούν την απώλεια αλλάζουμε την αναπαράσταση της συνάρτηση λάθους και θεωρούμε ότι οι άγνωστοι παράμετροι από τους οποίους επηρεάζεται είναι τα βάρη και η κλίση του ταξινομητή. Δηλ. αν θεωρήσουμε ότι το άνυσμα θ έχει τις τιμές του ανύσματος w ενσωματωμένης της τιμής b τότε έχουμε την συνάρτηση $L_{CE}(f(x^{(i)}; \theta), y^{(i)})$. Η συνάρτηση $f(x^{(i)}; \theta) = \hat{y}$ αντιπροσωπεύει την υπολογισμένη τιμή που τώρα έχουμε ενσωματώσει τα βάρη και την κλίση ως παραμέτρους της συνάρτησης. Ψάχνουμε να βρούμε τις τιμές του ανύσματος θ που θα μας δώσει την μικρότερη τιμή στην συνάρτηση απώλειας L_{CE}. Ο υπολογισμός των τιμών γίνεται σε ομάδες και όχι σε μονάδες. Η εξίσωση που εκφράζει τα παραπάνω είναι:

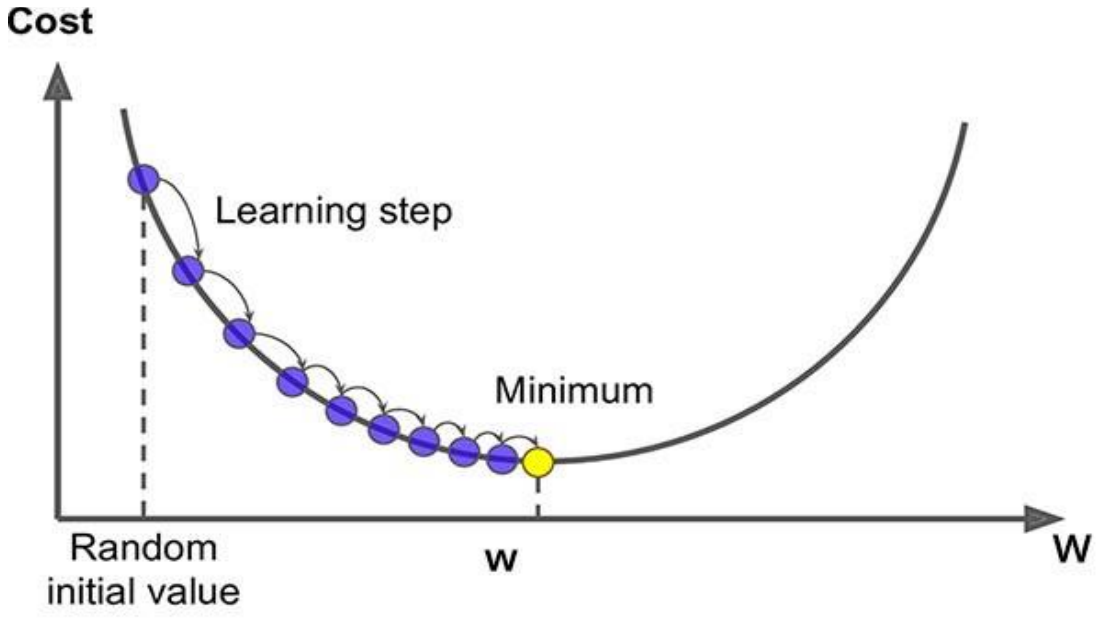
$$\hat{\theta} = \underset{\theta}{argmin} \frac{1}{m} \sum_{i=1}^{m} L_{CE}(f(x^{(i)}; \theta), y^{(i)}) \quad (12)$$

Η παραπάνω εξίσωση εκφράζει την αναζήτηση του ανύσματος θ με την αναζήτηση της καλύτερης εκτίμησης $\hat{\theta}$ η οποία εκφράζει την ελάχιστη παράμετρό της συνάρτησης $argmin$. Η **κάθοδος κλίσης** είναι μία μέθοδος για τον υπολογισμό της μικρότερης τιμής. Μία γραφική αναπαράσταση της μεθόδου για μία μόνο παράμετρο φαίνεται στο παρακάτω σχήμα. Βλέπουμε την πορεία που μπορεί να ακολουθήσει ο υπολογισμός της εκτιμώμενης τιμής $\hat{\theta}$ με την προϋπόθεση ότι η αναπαράσταση της συνάρτηση είναι κυρτή καμπύλη. Η μέθοδος δηλ. προσπαθεί να υπολογίσει την κλίση της συνάρτησης L_{CE} και να κινηθεί προς το κάτω μέρος της κλίσης. Το μήκος του βήματος μετακίνησης σε κάθε βήμα υπολογισμού εξαρτάται από έναν παράγοντα που ονομάζεται **ρυθμός μάθησης** η (**learning rate**). Στόχος είναι να προσεγγίσουμε το κάτω μέρος της καμπύλης που εκφράζει την μικρότερη τιμή απώλειας και κατά συνέπεια αποτυπώνει την καλύτερη εκτίμηση της θ που στην απλή περίπτωση που εξετάζουμε εκπροσωπεί μία μεταβλητή βάρους w.

Για τον υπολογισμό της κλίσης χρησιμοποιούμε την παράγωγο της συνάρτησης $\frac{d}{dw} f(x; w)$. Όπως βλέπουμε η παράγωγος γίνεται με βάση την τιμή του βάρους w. Η παράγωγος υπολογίζει την κλίση της συνάρτησης f στο συγκεκριμένο σημείο. Η κίνηση που πρέπει να κάνουμε είναι αντίθετη από την τιμή της παραγώγου πολλαπλασιασμένη με τον ρυθμό μάθησης η. Η εξίσωση που υπολογίζει τη νέα τιμή του w είναι:

$$w^{t+1} = w^t - \eta \frac{d}{dw} f(x; w)$$

Η διαδικασία που παρουσιάσαμε αφορούσε μία μεταβλητή βάρους που σημαίνει ότι έχουμε ένα μόνο χαρακτηριστικό. Η γραφική αναπαράσταση της συνάρτησης γίνεται σε χώρο δύο διαστάσεων. Στα πραγματικά όμως προβλήματα που αντιμετωπίζουμε



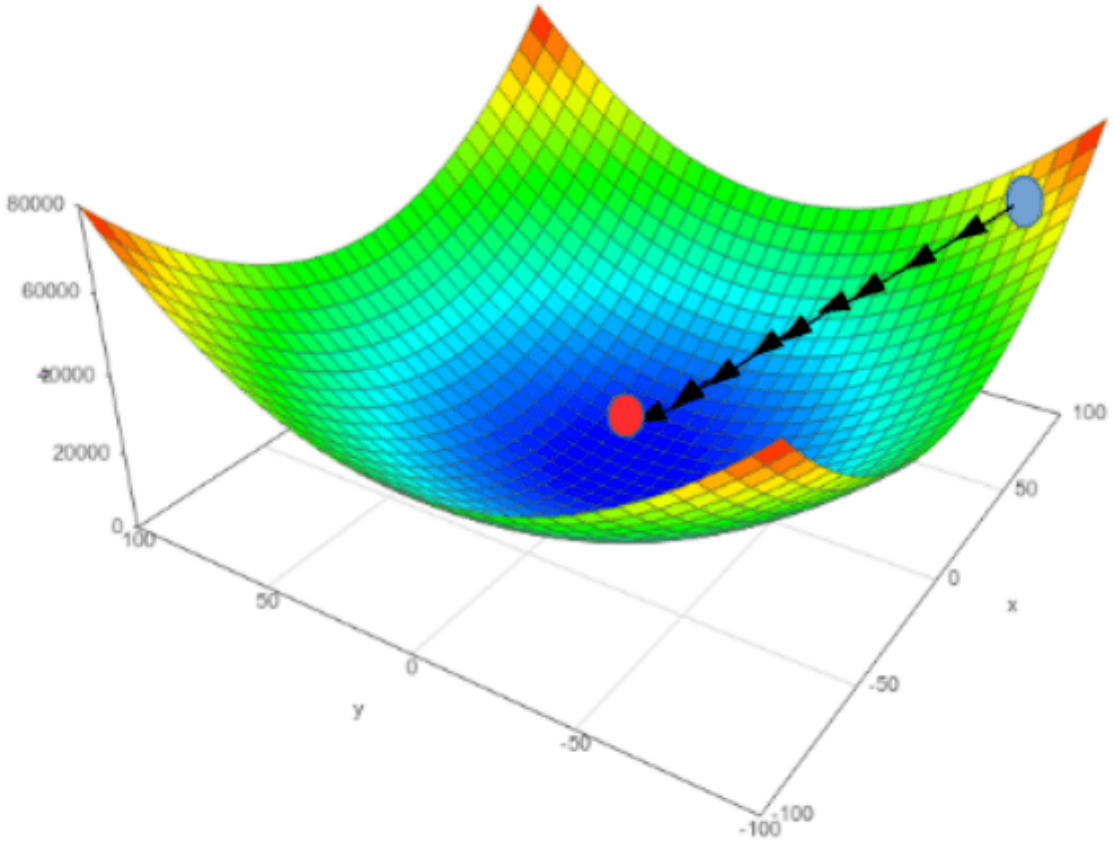
Διάγραμμα 1 Καμπύλη της συνάρτησης απώλειας

Η διαδικασία που παρουσιάσαμε αφορούσε μία μεταβλητή βάρους που σημαίνει ότι έχουμε ένα μόνο χαρακτηριστικό. Η γραφική αναπαράσταση της συνάρτησης γίνεται σε χώρο δύο διαστάσεων. Όπως είδαμε προηγουμένως έχουμε για πολλά χαρακτηριστικά και κατά συνέπεια για πολλά βάρη τα οποία συνθέτουν το άνυσμα στο οποίο έχουμε αναφερθεί. Αυτό μας μεταφέρει σε ένα πολυδιάστατο χώρο που η γεωμετρική του αναπαράσταση είναι δύσκολη. Μία προσέγγιση σε τρισδιάστατο χώρο εμφανίζεται στο επόμενο διάγραμμα (Διάγραμμα 2). Ο υπολογισμός της παραγώγου στην περίπτωση αυτή γίνεται στο άνυσμα των βαρών. Η κίνηση προς το κάτω μέρος της καμπύλης που είδαμε προηγουμένως με την μεταβολή της τιμής μίας εξαρτημένης μεταβλητής έχει αναχθεί σε μετακίνηση επιφάνειας πολλών διαστάσεων με μεταβολή πολλών βαρών. Η εξίσωση υπολογισμού της παραγώγου των ανυσμάτων παρουσιάζεται παρακάτω.

$$\nabla_\theta L(f(x;\theta),y) = \begin{bmatrix} \frac{d}{dw_1} L(f(x;\theta),y) \\ \cdots \\ \frac{d}{dw_n} L(f(x;\theta),y) \end{bmatrix}$$

Και η εξίσωση που υπολογίζει τις νέες τιμές βαρών είναι:

$$\theta_{t+1} = \theta_t - \eta \nabla L(f(x;\theta),y)$$



Διάγραμμα 2 Συνάρτηση απώλειας με πολλές μεταβλητές

Συστάδες με τον αλγόριθμο K-Means με χρήση του πολυμορφικού λεξικού συχνοτήτων

Ο K-Means είναι ο γνωστότερος ομαδοποιητής και ανήκει στην κατηγορία των μη εποπτευόμενων ταξινομητών. Η λειτουργία του είναι να διαμοιράζει τα δεδομένα, στην περίπτωσή μας κείμενα, σε ομάδες με όμοια χαρακτηριστικά. Ο τρόπος που το κάνει είναι πρώτα να απεικονίζει τα κείμενα σε ανύσματα αριθμών και στην συνέχεια να βρει τα κέντρα βάρους των ομάδων ώστε τα κείμενα της ομάδας να είναι σε κοντινή απόσταση. Χρησιμοποιείται στην πρώιμη φάση ανάλυσης όπου εξερευνούμε το προφίλ των δεδομένων μελετώντας τις ομάδες που αυτά δημιουργούν. Βασική προϋπόθεση για την εφαρμογή του αλγόριθμου είναι η αναπαράσταση του κειμένου ως σημείο σε ένα πολυδιάστατο χώρο. Το όνομα K-Means περιέχει τα δύο συστατικά του αλγόριθμου, 1) το K καθορίζει τον αριθμό των ομάδων και 2) Το Means που σηματοδοτεί τον τρόπο που καθορίζεται το κέντρο της ομάδας που γίνεται με την μέση τιμή των αποστάσεων όλων των σημείων της ομάδας.

Ο αλγόριθμος είναι μια επαναλαμβανόμενη διαδικασία όπου σε κάθε κύκλο υπολογίζουμε αν θα υπάρξει και νέα επανάληψη. Η συνθήκη επανάληψης καθορίζεται από τον αριθμό των αλλαγών ομάδας που έγιναν κατά την διάρκεια της επανάληψης. Σε κάθε επανάληψη επαναπροσδιορίζεται η ομάδα κάθε κειμένου. Αν δεν υπάρξει μεταβολή ή ο αριθμός των μεταβολών είναι μικρός η διαδικασία θα σταματήσει. Περιγραφικά τα βήματα του αλγόριθμου είναι:

1. Τυχαία επιλογή K σημείων από τα σημεία των κειμένων.
2. Επαναλαμβάνουμε τα επόμενα βήματα έως ότου δεν υπάρχουν αλλαγές

- Υπολογίζουμε την απόσταση κάθε σημείου κειμένου από κάθε κέντρο ομάδας με υπολογισμό της Ευκλείδειας απόστασης.
- Επιλογή του κοντινότερου κέντρου και μέσω αυτού της ομάδας που ανήκει.
- Υπολογισμός του κέντρου κάθε ομάδας με υπολογισμός της μέσης τιμής των σημείων της ομάδας.

Για την υλοποίηση του αλγόριθμου K-Means σε μία συλλογή εγγράφων χρησιμοποιούμε το λεξικό συχνοτήτων που παρουσιάσαμε παραπάνω. Το λεξικό αυτό έχει την απαραίτητη πληροφορία που χρειάζεται ο αλγόριθμος. Τα έγγραφα έχουν μετατραπεί σε πίνακες λεκτικών μονάδων και συχνοτήτων, δηλ. ζευγών αριθμών και είναι πλέον εύκολο να κάνουμε αριθμητικές πράξεις. Το μέγεθος του λεξιλόγιο ορίζει τις διαστάσεις του χώρου που θα απεικονιστούν τα έγγραφα και το σύνολο ζευγών (λεκτική μονάδα, συχνότητα) που περιέχει κάθε έγγραφο καθορίζει το σημείο του εγγράφου στον χώρο. Η απόσταση δύο σημείων, δηλ. του σημείου q που αντιστοιχεί στο έγγραφο και του p που είναι ένα από τα K κέντρα, εκφράζεται από την Ευκλείδεια απόσταση και δίνεται από τον παρακάτω τύπο που αποτελεί επέκταση του Πυθαγόρειου θεωρήματος. Το N είναι το μέγεθος του λεξιλογίου άρα ο αριθμός των διαστάσεων του χώρου.

$$\sqrt{\sum_{i=1}^{N}(q_i - p_i)^2}$$

Εύρεση ιεραρχίας συστάδων (Hierarchical clustering) με χρήση του πολυμορφικού λεξικού συχνοτήτων

Ο αλγόριθμος της ιεραρχικής συσταδοποίησης[6] εμφανίζεται με δύο παραλλαγές. Στην πρώτη ξεκινάει με μία συστάδα που περιέχει όλα τα έγγραφα και με συνεχείς διαιρέσεις συστάδων δημιουργεί τις τελικές. Στην δεύτερη παραλλαγή θεωρεί ότι έχουμε τόσες συστάδες όσα και τα έγγραφα και με συνεχείς ενοποιήσεις συστάδων καταλήγει στην τελική κατανομή ή σε μία συστάδα. Το όνομα προκύπτει από την οπτικοποίηση της διαδικασίας όπου μοιάζει με μία δεντρική δομή και για το λόγο αυτό ονομάζεται και δεντρόγραμμα.

Και στην περίπτωση του ιεραρχικής συσταδοποίησης χρειαζόμαστε ένα μέτρο που να μας δίνει την εγγύτητα δύο εγγράφων ή ενός εγγράφου και του κέντρου μιας συστάδας. Στην δική μας υλοποίηση χρησιμοποιούμε ως μέτρο την ομοιότητα συνημίτονου (cosine similarity[7]). Η ομοιότητα συνημίτονου παίρνει το όνομά της επειδή χρησιμοποιεί την γωνία που σχηματίζουν δύο σημεία στον χώρο όταν πάρουμε τις ευθείες από την αρχή των αξόνων. Όσο μικρότερη είναι η γωνία τόσο εγγύτερα θεωρούνται τα σημεία. Το συνημίτονο μίας γωνίας 0 μοιρών είναι το 1 και όσο μεγαλύτερη είναι η γωνία τόσο η τιμή του συνημίτονου μικραίνει. Αυτό προσφέρει ένα πλεονέκτημα στο μέτρο γιατί η τιμή έχει και μία στοχαστική σημασία μιας και στις πιθανότητες το 1 εκφράζει την βεβαιότητα να συμβεί κάτι και το 0 το απίθανο να συμβεί. Και εδώ χρησιμοποιούμε τα λεξικά συχνοτήτων όπου το μέγεθος του λεξιλογίου εκφράζει τις διαστάσεις του χώρου και τα έγγραφα είναι σημεία του χώρου αυτού. Το μέτρο που υπολογίζει το συνημίτονο δύο σημείων του πολυδιάστατου αυτού χώρου δίνεται από τον παρακάτω τύπο.

Θεωρούμε ότι έχουμε δύο έγγραφα τα A, B και το λεξιλόγιο έχει μέγεθος N.

$$\cos\theta = \frac{\sum_{i=1}^{N} A_i B_i}{\sqrt{\sum_{i=1}^{N} A_i^2 \sum_{i=1}^{N} B_i^2}}$$

Ο αλγόριθμος για τον υπολογισμό των συστάδων είναι μία επαναληπτική διαδικασία με τα παρακάτω βήματα. Έστω ότι έχουμε N λέξεις, M έγγραφα και θέλουμε να πάρουμε K συστάδες.

1. Ξεκινάμε με M συστάδες όσα και τα έγγραφα

[6] https://en.wikipedia.org/wiki/Hierarchical_clustering
[7] https://en.wikipedia.org/wiki/Cosine_similarity

2. Υπολογίζουμε την απόσταση κάθε εγγράφου με τα υπόλοιπα έγγραφα που δεν ανήκουν στην ίδια συστάδα.
3. Επιλέγουμε τα έγγραφα με την μικρότερη απόσταση και ενοποιούμε τις συστάδες τους.
4. Αν έχουμε φτάσει στις συστάδες K σταματάμε.

Και εδώ χρησιμοποιούμε το πολυμορφικό λεξικό συχνοτήτων αφού η πληροφορία που χρειαζόμαστε για τον υπολογισμό του αλγόριθμου υπάρχει στο λεξικό. Χρειαζόμαστε την απεικόνιση του εγγράφου στον χώρο των N διαστάσεων και την δυνατότητα να υπολογίσουμε το μέτρο της ομοιότητας δύο εγγράφων. Όπως και στην περίπτωση των συστάδων με τον αλγόριθμο K-Means δεν χρησιμοποιούμε το ευρετήριο του λεξικού αλλά μόνο τους πίνακες των εγγράφων με τα ζεύγη (λεκτική μονάδα, συχνότητα στο έγγραφο). Στην επόμενη ενότητα θα δούμε ένα από τους κλασσικούς αλγόριθμους αναζήτησης σε μία συλλογή εγγράφων με βάση μία ή περισσότερες λέξεις.

Αναζήτηση εγγράφων με χρήση του αλγόριθμου TFIDF (MB25)

Η στοχαστική αναζήτηση μίας συλλογής κειμένων με βάση ένα μικρό κομμάτι πληροφορίας αποτελεί ένα από τα κλασικά και καθοριστικά προβλήματα στον χώρο της πληροφορικής. Η γιγάντωση εταιρειών όπως η Yahoo και η Goggle με τεχνολογίες αιχμής αντιμετώπισης του προβλήματος δείχνει την σπουδαιότητα του. Το πρόβλημα μπορεί να διατυπωθεί ως εξής: Έχουμε μία συλλογή εγγράφων και θέλουμε να κάνουμε αναζήτηση με βάση μία ή περισσότερες λέξεις. Θέλουμε να βρούμε όχι μόνο ποια έγγραφα έχουν μία ή περισσότερες από τις λέξεις αναζήτησης αλλά να ταξινομήσουμε την λίστα των εγγράφων με τα ποιο «σημαντικά» στην κορυφή. Πως όμως μπορούμε να καθορίσουμε ποιο είναι σημαντικότερο κείμενο για μία λέξη που περιέχουν. Ο τύπος TF-IDF[8] και οι παραλλαγές του έδωσε μία ικανοποιητική λύση στο πρόβλημα η οποία κυριαρχεί ακόμα στον χώρο της Ανάκτησης Πληροφορίας (Information Retrieval).
Το όνομα του προέρχεται από τα αρχικά Term Frequency (Συχνότητα Όρου) – Αντίστροφη Συχνότητα Εγγράφου (Inverse Document Frequency) και θα περιγράψουμε σύντομα τις βασικές αρχές του και πως αξιοποιείται στο Mnemosyne.
Για να ταξινομήσουμε τα έγγραφα με βάση μία λέξη που περιέχουν, πρέπει να υπολογίσουμε το βάρος που έχει η λέξη για το έγγραφο. Ο τύπος TF-IDF δίνει το βάρος που έχει μία λέξη για ένα έγγραφο μίας συλλογής. Ο τύπος χρησιμοποιεί τις έννοιες του όρου (term) και έγγραφο (document) θεωρώντας ότι ένα έγγραφο είναι ένα δοχείο από όρους. Στην περίπτωσή του Mnemosyne οι όροι μπορεί να είναι λέξεις, λήμματα, ή q-grams σε κανονικοποιημένη (όλα κεφαλαία και άτονα) ή κανονική μορφή. Ο τύπος του TF-IDF είναι το γινόμενο δύο παραγόντων, του TF (Term Frequency – Συχνότητα Όρου) και του IDF (Inverse Document Frequency – Αντίστροφη Συχνότητα Εγγράφου). Οι τύποι που ορίζουν τις ποσότητες TF, IDF και TF-IDF είναι:

$$tf(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$$

Ο παραπάνω τύπος μας λέει ότι η ποσότητα $tf(t,d)$, όπου το t είναι ένας όρος και το d ένα έγγραφο είναι η συχνότητα του όρου στο έγγραφο διαιρούμενη με το σύνολο των όρων που περιέχει το έγγραφο. Καθορίζει πόσο σημαντικός είναι ο όρος για το έγγραφο. Όροι που εμφανίζονται πολλές φορές έχουν μεγαλύτερο βάρος για το κείμενο από ότι όροι που εμφανίζονται λίγες φορές. Η ποσότητα $idf(t,D)$ μας δίνει το βάρος που έχει ο όρος t για την συλλογή D η οποία έχει N έγγραφα.

$$idf(t,D) = \log\frac{N}{f(t,D)}$$

Η $f(t,D)$ ορίζεται ως $f(t,D) = |\{d \in D : t \in d\}|$ και εκφράζει τον αριθμό των εγγράφων της συλλογής D που περιέχουν τον όρο t. Ο τύπος για την ποσότητα IDF μας λέει ότι ένας όρος έχει μεγάλο βάρος στην συλλογή αν εμφανίζεται σε λίγα έγγραφα, δηλ. ο παρονομαστής είναι πολύ μικρότερος του αριθμητή. Αν όμως ένας όρος εμφανίζεται σε πολλά έγγραφα τότε το βάρος του μειώνεται μιας και ο παρονομαστής μεγαλώνει. Ο τελικός τύπος που ακολουθεί υπολογίζει το βάρος που έχει ένας όρος για ένα έγγραφο μίας συλλογής.

[8] https://en.wikipedia.org/wiki/Tf–idf

$$tfidf(t,d,D) = tf(t,d) \cdot idf(t,D)$$

Τα συστήματα Ανάκτησης Πληροφορίας χρησιμοποιούν το μέτρο του TFIDF ως εξής. Ευρετηριάζουν μια συλλογή εγγράφων D και προσφέρουν την λειτουργία της αναζήτησης στην συλλογή. Στην συλλογή αυτή παρέχουν την δυνατότητα αναζήτησης με βάση μία ή περισσότερες λέξεις. Από το ευρετήριο βρίσκουν ποια έγγραφα έχουν τις λέξεις και με χρήση της παραπάνω φόρμουλας ταξινομούν τα έγγραφα με βάση την ποσότητα που τους δίνει ο τύπος TF-IDF.

Στο Mnemosyne χρησιμοποιούμε μία παραλλαγή του TF-IDF που έχει δίνει καλύτερα αποτελέσματα, την Okapi BM25[9]. Ο τύπος που δίνει το βάρος για ένα όρο t είναι:

$$bm25(t,d,D) = \frac{f(t,d) \cdot (k_1 + 1)}{f(t,d) + k_1 \cdot (1 - b + b \cdot \frac{|d|}{avgdl})} \cdot idf_{bm25}(t,D)$$

Οι παράμετροι k_1 και b είναι στατιστικές ποσότητες που επηρεάζουν την κατανομή και κατά συνέπεια την ακρίβεια του τύπου. Η ποσότητα $idf_{bm25}(t,D)$ εκφράζει την Αντίστροφη Συχνότητα Εγγράφου. Μπορούμε να χρησιμοποιήσουμε τον τύπο που παρουσιάσαμε παραπάνω. Μια ενναλακτική φόρμουλα είναι:

$$idf_{bm25}(t,D) = \ln\left(\frac{N - f(t,D) + 0.5}{f(t,D) + 0.5} + 1\right)$$

Στο Mnemosyne η υλοποίηση του μηχανισμού BM25 βασίζεται στα λεξικά συχνοτήτων που παρουσιάσαμε παραπάνω. Τα λεξικά αυτά είναι αποθήκες οντοτήτων όπου η κάθε οντότητα αποτελείται από μία ή περισσότερες λεκτικές μονάδες. Τα έγγραφα δηλ. στο Mnemosyne είναι οντότητες και οι όροι εκφράζονται με τις λεκτικές οντότητες. Ως οντότητες μπορεί να έχουμε πολυλεκτικούς όρους, ονοματικά σύνολα (ανθρώπων, οργανισμών, τοπωνυμίων, κ.α.) αλλά και κανονικά κείμενα. Ως λεκτικές μονάδες μπορούμε να έχουμε λέξεις, λήμματα αλλά και τριγράμματα, τετραγράμματα, κ.λπ. Τα λεξικά έχουν όλη την πληροφορία που χρειαζόμαστε για την ανάκτηση των οντοτήτων. Έχουμε το ευρετήριο που μας καθορίζει ποιες οντότητες περιέχουν μία λεκτική μονάδα και τους πίνακες συχνοτήτων που μας επιτρέπουν να υπολογίσουμε τις παραπάνω οντότητες.

Ο μηχανισμός TF-IDF χρησιμοποιείται σε πολλές εφαρμογές ταυτοποίησης οντοτήτων και εύρεσης σχετικών κειμένων σε μία συλλογή. Το μειονέκτημά του είναι η στατικότητα του λεξικού το οποίο κτίζεται από την συλλογή πριν την αξιοποίησή του. Ως αντιστάθμισμα έχουμε την μεγάλη απόδοση και το περιορισμένο μέγεθος της συλλογής που μπορεί να μεταφερθεί από έναν υπολογιστή σε κάποιον άλλο.

Word2vec lexicon

Η γλώσσα αποτελείται από συνθέσεις συμβόλων όπως χαρακτήρων, λέξεων, προτάσεων, παραγράφων, κ.λπ. Η αξιοποίηση τεχνολογιών Μηχανικής Μάθησης και Στατιστικής στην Επεξεργασία Φυσικής Γλώσσας απαιτεί την αναπαράσταση των συμβόλων σε αριθμούς. Είδαμε παραπάνω την περίπτωση όπου οι λεκτικές μονάδες εκπροσωπούνται με έναν αριθμό που είναι η θέση της λεκτικής μονάδας στο ευρετήριο ενός πολυμορφικού λεξικού. Επίσης είδαμε πως αναπαριστούμε ένα έγγραφο με ένα σύνολο ζευγών (λεκτική μονάδα, συχνότητα) και το απεικονίζουμε σε ένα σημείο σε ένα πολυδιάστατο χώρο με διαστάσεις το μέγεθος του λεξιλογίου.

Το πρόβλημα με την απεικόνιση των λεκτικών μονάδων σε αριθμούς με βάση την θέση τους στο ευρετήριο είναι ότι η επιλογή των αριθμών είναι «σχεδόν» τυχαία και δεν απεικονίζουν καμία ιδιότητα της λεκτικής μονάδας στην συλλογή εγγράφων. Η χρήση των αριθμών είναι επικεντρωμένη στην ύπαρξη ή όχι της λεκτικής μονάδας και της συχνότητας της και δεν μπορούμε να εξάγουμε καμία πληροφορία για τον τρόπο που η λεκτική μονάδα χρησιμοποιείται αλλά και τα συγκείμενά της.

Η προσπάθεια να βρεθεί μια καλύτερη αριθμητική αναπαράσταση των λέξεων οδήγησε στην αναζήτηση αναπαράστασης με άνυσμα αριθμών που αντιπροσωπεύει ένα σημείο σε πολυδιάστατο χώρο. Σε αντιπαράθεση, η αναπαράσταση της λέξης με ένα αριθμό μπορεί να απεικονίσει την λέξη σε ένα σημείο ενός μονοδιάστατου χώρου. Οι σχέσεις των λέξεων που συνωστίζονται σε κείμενα της συλλογής δεν

[9] https://en.wikipedia.org/wiki/Okapi_BM25

μπορούν να αναπαρασταθούν σε αυτή την απεικόνιση. Χρειαζόμαστε περισσότερες διαστάσεις για να ενσωματώσουν ιδιότητες χρήσης της λέξης στην συλλογή.

Μία από τις μεθόδους για την κατασκευή αντιπροσωπευτικών ανυσμάτων, χρησιμοποιεί πίνακες γειτνίασης. Οι διαστάσεις του πίνακα γειτνίασης είναι $n \cdot m$ όπου το n είναι το μέγεθος του λεξιλογίου και m ο αριθμός των συγκείμενων. Το συγκείμενο μπορεί να είναι το έγγραφο ή ένα παράθυρο l λέξεων σε μία πρόταση. Αν μία λέξη βρεθεί σε ένα συγκείμενο τότε τα αντίστοιχα κελιά θα έχουν μη μηδενικές τιμές. Ο αριθμός που έχει το κάθε κελί του πίνακα εκφράζει την συχνότητα εμφάνισης των δύο λέξεων. Μία παραλλαγή για την τιμή του κελιού είναι να χρησιμοποιήσουμε το μέτρο tfidf που παρουσιάσαμε παραπάνω.

Στο τέλος της δημιουργίας του πίνακα, κάθε λέξη αντιστοιχεί σε ένα άνυσμα m αριθμών που εκφράζουν τα συγκείμενα που έχει εμφανιστεί. Το άνυσμα αυτό είναι πολύ μεγάλο και αραιό και είναι δύσκολο να αξιοποιηθεί αποδοτικά. Για να μειωθεί το μήκος του εφαρμόζονται μέθοδοι όπως η μέθοδος SVD[10] ή η PCA[11] οι οποίες διασπούν τον πίνακα γειτνίασης σε μικρότερους σε διαστάσεις πίνακες. Η ιδέα των μεθόδων αυτών είναι να μειώσουν την διάσταση του πίνακα μέσω αναπαράστασης του ως γινόμενο δύο ή περισσότερων πινάκων (παραγοντοποίηση του πίνακα). Για παράδειγμα αν έχουμε τον πίνακα D με διαστάσεις $n \cdot m$ ψάχνουμε να βρούμε δύο πίνακες U και V έτσι ώστε να ισχύει ο παρακάτω τύπος.

$$D \approx U V^T$$

Οι διαστάσεις των U και V είναι αντίστοιχα $n \cdot k$ και $m \cdot k$ όπου το k είναι μία σχετικά μικρή ποσότητα. Στην περίπτωση αυτή το άνυσμα μίας λέξης από m διαστάσεις έχει πέσει στις k.

Τα τελευταία χρόνια το βάρος έχει πέσει στην έρευνα εξεύρεσης τεχνικών αξιοποίησης μεγάλων συλλογών χωρίς την ανθρώπινη συμμετοχή (μη-εποπτευόμενη μηχανική μάθηση) και έχει δώσει πολύ αποτελέσματα. Για παράδειγμα, οι μέθοδοι που είναι γνωστοί με το όνομα word2vec[12] έχουν πολύ καλά αποτελέσματα στην εκμάθηση διανυσματικών αναπαραστάσεων λέξεων από μεγάλες συλλογές. Τα διανύσματα αυτά δημιουργούν Κατανεμημένα Σημασιολογικά Μοντέλα (Distributional Semantic Models – DSM) τα οποία στηρίζονται στην ιδέα ότι λέξεις με παρόμοιο νόημα έχουν την τάση να εμφανίζονται σε παρόμοια συγκείμενα. Η «Κατανεμημένη Υπόθεση» (Distributional Hypothesis[13]) παρέχει το θεωρητικό υπόβαθρο στην κατανόηση και υπολογισμό των σημασιολογικών σχέσεων μεταξύ των λέξεων. Ανάλογα με το ποιο κομμάτι του συγκείμενου προσπαθούμε να προβλέψουν έχουμε διαφορετικά μοντέλα με τους αντίστοιχους αλγόριθμους εκπαίδευσης.

- Η CBOW (Continuous Bag Of Words – Συνεχόμενος Σάκος Από Λέξεις) προσπαθεί να προβλέψει μία λέξη από το συγκείμενο της.
- Η Skip Grams προσπαθεί να μαντέψει το συγκείμενο με βάση μία λέξη

Ένα παράδειγμα που δείχνει την ποιότητα των ανυσμάτων που παράγουν τα μοντέλα αυτά και χρησιμοποιείται και ως σημείο αναφοράς είναι η παρακάτω έκφραση θεωρώντας ότι η κάθε λέξη εκπροσωπείτε με ένα άνυσμα.

«βασιλιάς» - «βασίλισσα» ≈ «άνδρας» - «γυναίκα»

Δηλ. η απόσταση των σημείων που αντιπροσωπεύουν τις λέξεις «βασιλιάς» και «βασίλισσα» είναι σχεδόν ίση με την απόσταση των σημείων που αντιπροσωπεύουν τις λέξεις «άνδρας» και «γυναίκα». Αυτή η μαθηματική ισοδυναμία μας δίνει την δυνατότητα να κάνουμε σημασιολογικές ερωτήσεις της μορφής:

«βασιλιάς» - **X** ≈ «άνδρας» - «γυναίκα»

Δηλ. ποιες λέξεις απέχουν από την λέξη «βασιλιά» όσο απέχουν οι λέξεις «άνδρας» και «γυναίκα».

Οι παραπάνω ισοδυναμίες μπορούν να αξιοποιηθούν στην εύρεση συνωνύμων αλλά και μορφολογικών παραλλαγών μιας λέξης (π.χ. τοπική ή χρονική διαφοροποίηση μιας λέξης), να προβλέψουν μια ακολουθία λέξεων που έχει σβηστεί, να δημιουργήσει κείμενο, κ.α.

[10] https://en.wikipedia.org/wiki/Singular_value_decomposition
[11] https://en.wikipedia.org/wiki/Principal_component_analysis
[12] https://en.wikipedia.org/wiki/Word2vec
[13] https://en.wikipedia.org/wiki/Distributional_semantics

Σώμα Κειμένων

Για τον υπολογισμό των ανυσμάτων «ενσωμάτωσης λέξεων» (word embeddings) το Mnemosyne χρησιμοποιεί τα πρωτότυπα προγράμματα των κατασκευαστών. Έχουν δημιουργηθεί εργαλεία που λειτουργούν ως εντολοδόχοι των προγραμμάτων. Για την παραγωγή των ανυσμάτων χρειαζόμαστε ένα σώμα κειμένων μεγάλου όγκου και με ποικιλία περιεχομένου. Η συλλογή του σώματος που συλλέχθηκε για τις ανάγκες της δημιουργίας των μοντέλων word2vec έγινε με κείμενα από ειδησεογραφικές τοποθεσίες, βιβλία ανοικτού περιεχομένου, εκπαιδευτικά συγγράμματα, υπότιτλους κινηματογραφικών ταινιών, ιστολόγια, μέσα κοινωνικής δικτύωσης, κυβερνητικά έγγραφα από το εθνικό τυπογραφείο κ.α. Αξιοποιήθηκαν όλα τα ανοικτά δεδομένα που βρέθηκαν είτε σε στατική μορφή με καταβίβαση του περιεχομένου είτε σε δυναμική μορφή με προγράμματα εξερεύνησης των ιστότοπων και εξαγωγής του περιεχομένου τους. Το περιεχόμενο που συλλέχθηκε ήταν σε πολλές ηλεκτρονικές μορφές και με πολλά περιττά στοιχεία που έπρεπε να καθαρίσουν για να αξιοποιηθεί το καθαρό κείμενο. Για τις λειτουργίες της εκκαθάρισης και συγκέντρωσης του καθαρού κειμένου δημιουργήθηκαν ειδικά εργαλεία όπως το `DataPrep` που παρουσιάσαμε παραπάνω. Το τελικό καθαρό κείμενο του σώματος συγκεντρώθηκε σε ένα αρχείο και αποτέλεσε την είσοδο στην επεξεργασία των εργαλείων word2vec. Το συγκεντρωτικό αρχείο του σώματος είχε μέγεθος ~17G και είχε ~1.6G λέξεις. Τα μοντέλα που δημιουργήθηκαν ήταν 300 διαστάσεων και το μέγεθος του λεξιλογίου ~4.5M λέξεις.

Τα μοντέλα που παράγουν τα προγράμματα word2vec είναι σε μορφή κειμένου όπου η κάθε γραμμή έχει μία λέξη από το λεξιλόγιο ακολουθούμενη από 300 πραγματικούς αριθμούς. Για την αποδοτική αξιοποίηση των μοντέλων κτίζονται πολυμορφικά λεξικά συχνοτήτων όπως παρουσιάσαμε παραπάνω.

Για την αξιολόγηση των μοντέλων χρησιμοποιήθηκαν τρεις κατηγορίες από ποιοτικά τεστ.

1. Τα τεστ ομοιότητας όπου για κάθε λέξη του τεστ βρίσκαμε τις πιο κοντινές της λέξεις και ελέγχαμε την συνάφεια των αποτελεσμάτων.
2. Τα τεστ αναλογικότητας όπου χρησιμοποιήθηκαν τετράδες όπως το παράδειγμα που παρουσιάσαμε παραπάνω με τους (ΒΑΣΙΛΙΑΣ, ΒΑΣΙΛΙΣΣΑ, ΑΝΔΡΑΣ, ΓΥΝΑΙΚΑ). Άλλο ένα γνωστό παράδειγμα τετράδας είναι η (ΓΑΛΛΙΑ, ΠΑΡΙΣΙ, ΕΛΛΑΔΑ, ΑΘΗΝΑ)
3. Τα τεστ ακραίων/άσχετων λέξεων όπου για μια ομάδα λέξεων βρίσκαμε την πιο απομακρυσμένη από τις υπόλοιπες και η οποία έπρεπε να είναι και σημασιολογικά η πιο ξένη λέξη. Ένα παράδειγμα ομάδας για έλεγχο είναι η { ΝΟΣΤΙΜΟ, ΦΑΓΗΤΟ, ΠΡΩΙΝΟ, ΓΕΥΜΑ, ΒΡΑΔΙΝΟ, ΠΟΤΟ }

Για την χρήση των λεξικών υλοποιήθηκε ένας αριθμός από μεθόδους που αξιοποιούνται είτε ως βιβλιοθήκες από προγράμματα Java είτε από αναλυτές μέσα από κανόνες του φορμαλισμού «Κανών».

Αρχιτεκτονική Νέφους – Το Mnemosyne ως Microservice

Η εποχή των μονολιθικών εφαρμογών με πλούσια και σύνθετη λειτουργικότητα έχει αρχίσει να σβήνει και την θέση της παίρνουν οι κατανεμημένες εφαρμογές νέφους. Οι αρχιτεκτονικές νέφους προσφέρουν καλύτερη αξιοποίηση των πόρων, μεγάλη αξιοπιστία, καλύτερο έλεγχο και παρακολούθηση, μεγαλύτερη ασφάλεια, αδιάλειπτη λειτουργία, γρήγορη αντικατάσταση και πλήθος άλλων πλεονεκτημάτων. Ένα σημαντικό χαρακτηριστικό των σύγχρονων αυτών τεχνολογιών είναι το χαμηλό κόστος το χαμηλό κόστος λειτουργίας και συντήρησης της εφαρμογής όταν μπει στο νέφος. Οι τεχνολογίες που ενσωματώνουν λειτουργίες που χρειάζονται οι εφαρμογές νέφους είναι ώριμες και με μικρό κόστος ενσωμάτωσης τους.

Στα πλαίσια της ενσωμάτωσης τεχνολογιών νέφους στο Mnemosyne έχουμε κάνει αρχιτεκτονικές αλλαγές στον κώδικα. Ο κώδικας του Mnemosyne έχει > 1000 κλάσεις οι οποίες υλοποιούν ένα μεγάλο αριθμό λειτουργιών όπως:

- Διεπαφές με Βάσεις Δεδομένων όπως οι SQL Server, MySQL, Postgres, AS400, κ.α. Οι λειτουργίες που πραγματοποιούνται σε αυτές τις διεπαφές είναι:
    - Επεξεργασία πληροφορίας οντοτήτων όπως πρόσωπα, διευθύνσεις, οργανισμοί για το κτίσιμο λεξικών που ενσωματώνουν την εταιρική γνώση (πόρους) και που θα χρησιμοποιηθούν αργότερα στις αναλύσεις.

    - Ενημέρωση πινάκων της Βάσης Δεδομένων με δομημένη πληροφορία που προέκυψε από την ανάλυση κειμένων. Αφορά την εξαγωγή πληροφορίας από κείμενα, την μετατροπή της πληροφορίας σε δομημένη και την αποθήκευσή της στην βάση του πελάτη.
    - Αποθήκευση των ενδιάμεσων αποτελεσμάτων των αναλύσεων. Κατά την διάρκεια της ανάλυσης μίας συλλογής παράγονται αποτελέσματα μεγάλου όγκου τα οποία μπορούν να αποθηκευτούν σε αρχεία με μορφή XML ή Json ή σε βάση δεδομένων.
- Βιβλιοθήκες ανάγνωσης και εξαγωγής κειμένου από έγγραφα μορφές όπως (.txt, .xml, .json, .csv, .pdf, κ.α.).
- Μηχανισμούς επικοινωνίας με εξωτερικές πηγές όπως Twitter, Ιστολόγια, δικτυακές πύλες, κ.α.
- Εργαλεία κατασκευή λεξικών όπως έχουμε αναφέρει παραπάνω.
- Βοηθήματα για προετοιμασία και φιλτράρισμα δεδομένων σε όλες τις φάσεις ανάλυσης.
- Βιβλιοθήκες γλωσσικών εργαλείων για Ορθογραφικό Έλεγχο, Συλλαβισμό, Θησαυρό Συνωνύμων, Λημματοποιητή, Εύρεση Ορολογίας, Γραμματικό Διορθωτή.
- Τεχνολογίες εκκαθάρισης και ταυτοποίησης δεδομένων κειμενικής μορφής.
- Συνθέσεις αναλύσεων για την επεξεργασία συλλογών εγγράφων.

Εκτός από τις ενδεικτικές παραπάνω λειτουργίες, ο κώδικας εμπλουτίζεται συνεχώς με τεχνολογίες αιχμής από την συμμετοχή σε ερευνητικά έργα. Η διαχείρισή του σε μία μονολιθική αρχιτεκτονική δημιουργεί συχνά προβλήματα με συνέπεια να υπάρχουν πολλές εκδόσεις σε διαφορετικές εγκαταστάσεις.

Η εγκατάσταση και παραμετροποίηση μιας εγκατάστασης του Mnemosyne είναι μια εξειδικευμένη εργασία που δεν έχει αυτοματοποιηθεί σε ικανοποιητικό βαθμό. Η αξιοποίηση του από άλλες εφαρμογές και κυρίως από εφαρμογές ιστού απαιτούσε σύνδεσή του με αποδοτικούς και αξιόπιστους μηχανισμούς επικοινωνίας.

Οι τεχνολογίες νέφους έχουν απλοποιήσει τις παραπάνω λειτουργίες. Η εγκατάσταση μιας σύνθετης εφαρμογής απλοποιείται με την ενσωμάτωσή της σε ένα κιβώτιο (container) και την λειτουργίας του ως docker. Η επικοινωνία με εξωτερικές εφαρμογές πραγματοποιείται μέσα από RESTful[14] Web Services και μηνύματα με μορφή Json. Όλες οι παραπάνω λειτουργίες προσφέρονται από περιβάλλοντα όπως το Spring Boot[15] που έχουμε χρησιμοποιήσει. Τόσο η διαδικασία της ενσωμάτωσης των τεχνολογιών όσο και η δημιουργία και υλοποίηση διεπαφών με πρωτόκολλα http και μηνύματα json είναι απλοποιημένες και απαιτούν ελάχιστη επιβάρυνση σε κώδικα. Η δημιουργία του docker image ώστε να τρέχει μέσα σε ένα container γίνεται από εξωτερικά εργαλεία που τρέχουν στο περιβάλλον κτισίματος και εγκατάστασης εφαρμογών Maven[16]

Στην υπάρχουσα έκδοση έχουν υλοποιηθεί οι παρακάτω υπηρεσίες

- Υπηρεσία Ορθογραφικού Ελέγχου
- Υπηρεσία Θησαυρού Συνωνύμων
- Υπηρεσία Διερεύνησης λεξικών word2vec.

[14] https://en.wikipedia.org/wiki/Representational_state_transfer
[15] https://spring.io/projects/spring-boot
[16] https://maven.apache.org

3. Πηγές και εργαλεία

Το σύνολο των δεδομένων που περιέχουν οι πηγές που περιγράφηκαν παραπάνω, απέκτησαν επί πλέον χαρακτηριστικά και μεταδεδομένα για τις ανάγκες του έργου, με βάση και τα γραμματικά φαινόμενα (βλ. Π1.1) που καταγράφηκαν και επιλέγησαν στην αρχή του έργου.

Τα χαρακτηριστικά αυτά προέκυψαν από εμπλουτισμό μέσα από μια σειρά πηγών (όπως έχει περιγραφεί μέχρι τώρα).

Α) από το σώμα κειμένων που δημιουργήθηκε και αποτελείται από τα σχολικά βιβλία (http://ebooks.edu.gr/ebooks/) που συντηρεί (ως υλικό), παράγει και διανέμει το ΙΤΥΕ «Διόφαντος»,

Β) Από Λεξικά που έχουν τα σχολικά βιβλία και πιο συγκεκριμένα το Λεξικό Α-Β-Γ Δημοτικού (http://ebooks.edu.gr/ebooks/v/html/8547/2350/Lexiko_A-B-G-Dimotikou_html-apli/index.html), το ορθογραφικό Λεξικό Δ-Ε-ΣΤ Δημοτικού (http://ebooks.edu.gr/ebooks/v/html/8547/2354/Orthografiko-Lexiko_D-E-ST-Dimotikou_html-apli/) και το Ερμηνευτικό Λεξικό της Νέας Ελληνικής για την Α-Β-Γ Γυμνασίου (http://ebooks.edu.gr/ebooks/v/html/8547/2332/Ermineutiko-Lexiko-Neas-Ellinikis_A-B-G-Gymnasiou_html-apli/).

Γ) Από το περιεχόμενο του Φωτόδεντρου (http://photodentro.edu.gr/lor/), του Εθνικού Συσσωρευτή Μαθησιακών Αντικειμένων, που έχει υλοποιήσει και συντηρεί το ΙΤΥΕ (Λέξεις – κλειδιά, Θεματική Περιοχή, Εκπαιδευτική Βαθμίδα, Τυπικό εύρος ηλικίας, περιγραφές, κ.α.).

Δ) Από την Βικιπαίδεια (https://el.wikipedia.org/wiki/) και το υλικά που αυτή περιέχει (Θεματικές Ενότητες, Λεκτικά - Όροι, Περιγραφές, Φράσεις από αναζήτηση, κ.α.).

Ε) Από την Βιβλιοθήκη Εκπαιδευτικών Δραστηριοτήτων «Ιφιγένεια» (http://ifigeneia.cti.gr/repository/).

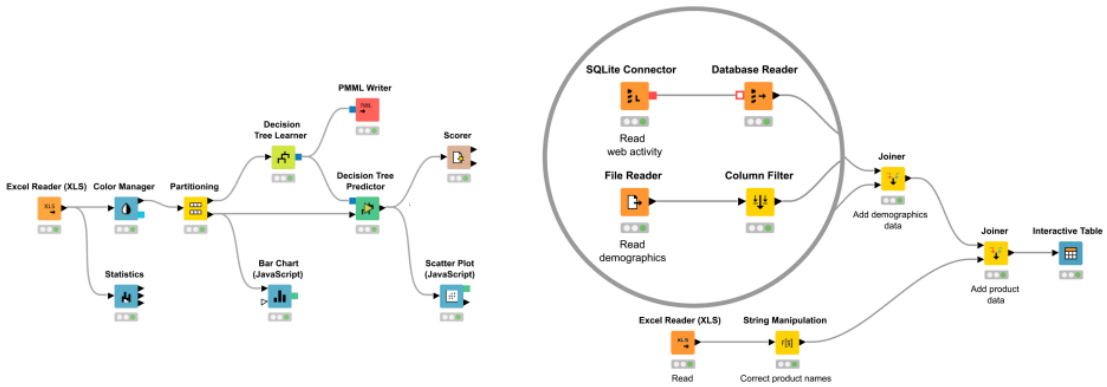
ΣΤ) Από την Τράπεζα Κειμένων του Κέντρου Ελληνικής Γλώσσας (https://www.greek-language.gr/certification/dbs/teachers/index.html) που είναι ελεύθερη προς χρήση και αξιοποίηση για διδακτικούς σκοπούς.

Για να γίνει δυνατή η επεξεργασία και διασύνδεση των παραπάνω πηγών με τα ήδη υπάρχοντα δεδομένα χρησιμοποιήθηκε η υποδομή KNIME που περιγράφεται εν συντομία στη συνέχεια.

Η Πλατφόρμα KNIME

Η πλατφόρμα KNIME Analytics (https://www.knime.com/knime-analytics-platform) είναι ένα λογισμικό ανοιχτού κώδικα, που βοηθά στο σχεδιασμό ροών εργασίας δεδομένων (workflows) και στην ανάπτυξη επαναχρησιμοποιήσιμων στοιχείων-συστατικών (components). Χρησιμοποιείται για την ανάλυση δεδομένων (data analytics), τη δημιουργία αναφορών (reporting) και την ενσωμάτωση (integration) διαφορετικών τεχνολογιών και ετερόκλητου λογισμικού. Δίνει τη δυνατότητα στους χρήστες να συνδυάσουν δεδομένα από οποιαδήποτε πηγή. Να επεξεργαστούν μορφές κειμένου όπως CSV, PDF, XLS, JSON, XML κ.α.. Να συνδεθούν σε πλήθος βάσεων δεδομένων όπως Oracle, Microsoft SQL, MySQL κ.α., σε αποθήκες δεδομένων όπως το Apache Hive, το Snowflake κ.α. και να ανακτούν δεδομένα από πηγές όπως το Salesforce, SharePoint, SAP Reader (Theobald Software), Twitter, AWS S3, Google Sheets, Azure και από άλλες κατανεμημένες εφαρμογές νέφους.

Στις παρακάτω εικόνες παρουσιάζονται ενδεικτικά δυο workflows με κόμβους που χρησιμοποιούν τις παραπάνω τεχνολογίες.



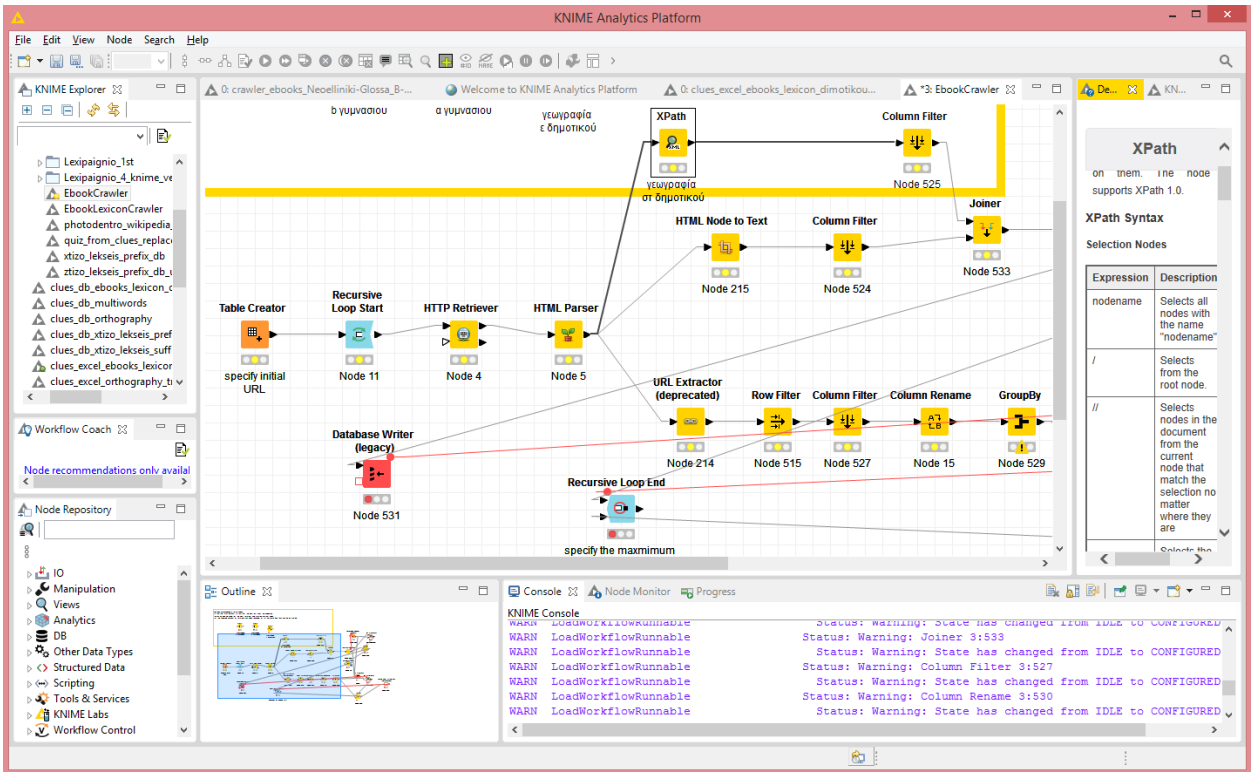
Στο παρόν έργο η πλατφόρμα Knime χρησιμοποιήθηκε για τις ανάγκες επεξεργασίας τύπου ETL (Extract, Transform, Load) στις διάφορες φάσεις του έργου, τη διασύνδεση με τρίτα συστήματα (εκτός πλατφόρμας Mnemosyne) και τον εμπλουτισμό των πόρων μέσα από τα κατάλληλα workflows που υλοποιήθηκαν. Επίσης χρησιμοποιήθηκε για την επικοινωνία με την πλατφόρμα Mnemosyne μέσω των RESTful Web Services και μηνυμάτων με μορφή json.

Συνοπτικά τα workflows που υλοποιήθηκαν στην πλατφόρμα Knime ομαδοποιούνται με βάσει το τι εξυπηρετούν ως εξής:

- Το αυτόματο κατέβασμα δεδομένων από τις πηγές που αξιοποιήθηκαν για το έργο και την αποθήκευσή τους σε βάση δεδομένων.
- Την εξαγωγή όρων/φράσεων με βάσει τη δομή των κειμένων των πηγών.
- Την εξαγωγή του κειμένου από τα pdf αρχεία του σώματος κειμένου.
- Τη δημιουργία των indexes των εξαγόμενων όρων του σώματος κειμένου.
- Την επεξεργασία των xml αρχείων με τα αποτελέσματα των κανόνων “Kanon” της πλατφόρμας Mnemosyne για το σώμα κειμένων.
- Τη διασύνδεση των πηγών με τους εξαγόμενους όρους των κειμένων τους που προέκυψαν ως αποτέλεσμα της εφαρμογής των κανόνων “Kanon” της πλατφόρμας Mnemosyne.
- Τη χρήση των web services της Βικιπαιδείας από όπου αντλήθηκε υλικό.
- Τη χρήση των web services της πλατφόρμας Mnemosyne για την επεξεργασία και χαρακτηρισμό των δεδομένων.

Στην παρακάτω εικόνα παρουσιάζεται ένα στιγμιότυπο από το IDE της πλατφόρμας Knime στο οποίο γίνεται η επεξεργασία ενός workflow για το crawling των ebooks σχολικών βιβλίων και την αποθήκευση των δεδομένων στη βάση δεδομένων.

Λεξιπαίγνιο: Π3.2 Αναφορά Τεκμηρίωσης



4. Παραδοτέο Π3.3

Το περιεχόμενο του Παραδοτέου Π.3.3: Χαρακτηρισμένο γλωσσικό υλικό, περιέχει το σύνολο του υλικού (δεδομένων και μεταδεδομένων) που έχει παραχθεί μέχρι στιγμής στο έργο. Στη συνέχεια Περιγράφονται τα περιεχόμενά του.

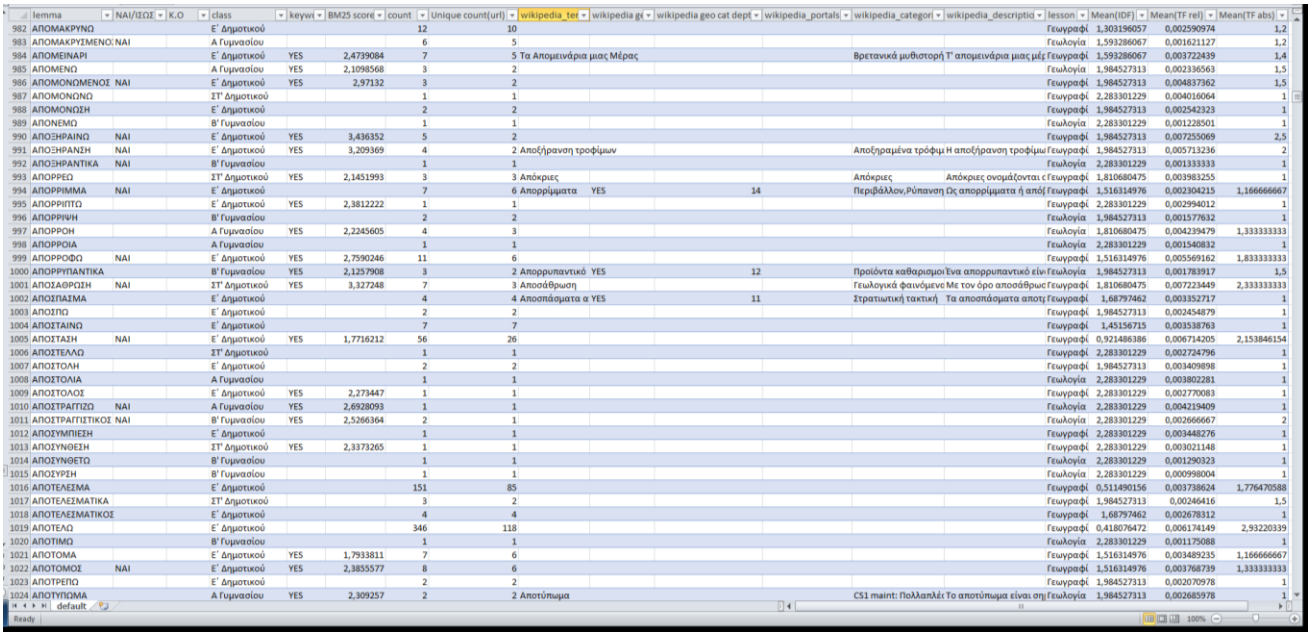
Αποτελέσματα επεξεργασίας δεδομένων (Αρχεία)

Για τις ανάγκες των παιχνιδιών, κατόπιν εστιασμένης επεξεργασίας των δεδομένων των πηγών που αναφέρθηκαν στο προηγούμενο κεφάλαιο, έχουν παραχθεί σε μορφή αρχείων, μια σειρά από αποτελέσματα τα οποία αφενός αποτελούν την είσοδο σε συγκεκριμένα παιχνίδια και από την άλλη μπορούν να επεξεργαστούν/αξιολογηθούν από τον καθηγητή πριν την χρήση τους. Ενδεικτικά παραδείγματα παρατίθενται στη συνέχεια, ενώ το σύνολο των σχετικών αρχείων περιέχεται στο Παραδοτέο Π3.3, κάτω από την ετικέτα «αποτελέσματα επεξεργασίας δεδομένων). Η περιγραφή των σχετικών αρχείων, περιλαμβάνεται στο Παράρτημα 1.2.

Ενδεικτικά, στο κεφάλαιο αυτό, αναφέρονται τα:

Πολυλεκτικοί όροι από τα βιβλία της γεωγραφίας

Το συγκεκριμένο αρχείο (excel - ebooks_lemma_terms_with_wikipedia_info_v8.xlsx) περιέχει «Πολυλεκτικτούς όροους» από τα σχολικά βιβλία της Γεωγραφίας με ένδειξη για κύρια ονόματα. Έχουν αναζητηθεί στην Βικιπαιδεία και υπάρχει πληροφορία από αυτήν. Είναι συνδεδεμένοι με τις λέξεις - κλειδιά και περιέχουν πληροφορία για την συχνότητα εμφάνισής τους στα βιβλία.

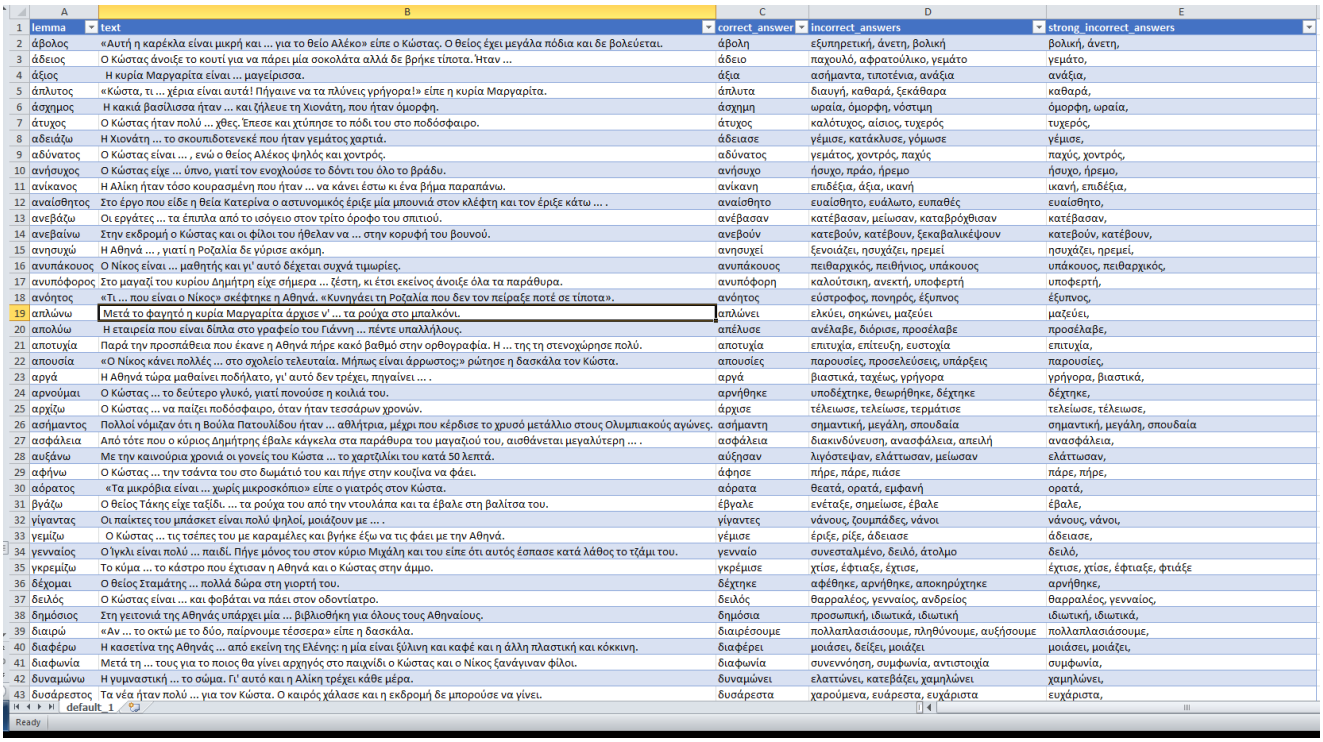


	lemma	ΝΑΙ/ΙΣΩΣ	Κ.Ο	class	keywi	BM25 score	count	Unique count(url)	wikipedia_tei	wikipedia g	wikipedia geo cat dept	wikipedia_portals	wikipedia_categor	wikipedia_descriptio	lesson	Mean(IDF)	Mean(TF rel)	Mean(TF abs)
982	ΑΠΟΜΑΚΡΥΝΩ			Ε΄ Δημοτικού			12	10							Γεωγραφί	1,303196057	0,002590974	1,2
983	ΑΠΟΜΑΚΡΥΣΜΕΝΟ	ΝΑΙ		Α Γυμνασίου			6	5							Γεωλογία	1,593286067	0,001621127	1,2
984	ΑΠΟΜΕΙΝΑΡΙ			Ε΄ Δημοτικού	YES	2,4739084	7	5	Τα Απομεινάρια μιας Μέρας				Βρετανικά μυθιστορή	Τ' απομεινάρια μιας μέ	Γεωγραφί	1,593286067	0,003722439	1,4
985	ΑΠΟΜΕΝΩ			Α Γυμνασίου	YES	2,1098568	3	2							Γεωλογία	1,984527313	0,002336563	1,5
986	ΑΠΟΜΟΝΩΜΕΝΟΣ	ΝΑΙ		Ε΄ Δημοτικού	YES	2,97132	3	2							Γεωγραφί	1,984527313	0,004837362	1,5
987	ΑΠΟΜΟΝΩΝΩ			ΣΤ΄ Δημοτικού			1	1							Γεωγραφί	2,283301229	0,004016064	1
988	ΑΠΟΜΟΝΩΣΗ			Ε΄ Δημοτικού			2	2							Γεωγραφί	1,984527313	0,002542323	1
989	ΑΠΟΝΕΜΩ			Β΄ Γυμνασίου			1	1							Γεωλογία	2,283301229	0,001228501	1
990	ΑΠΟΞΗΡΑΙΝΩ	ΝΑΙ		Ε΄ Δημοτικού	YES	3,436352	5	2							Γεωγραφί	1,984527313	0,007255069	2,5
991	ΑΠΟΞΗΡΑΝΣΗ	ΝΑΙ		Ε΄ Δημοτικού	YES	3,209369	4	2	Αποξήρανση τροφίμων				Αποξηραμένα τρόφιμ	Η αποξήρανση τροφίμω	Γεωγραφί	1,984527313	0,005713236	2
992	ΑΠΟΞΗΡΑΝΤΙΚΑ	ΝΑΙ		Β΄ Γυμνασίου			1	1							Γεωλογία	2,283301229	0,001333333	1
993	ΑΠΟΡΡΕΩ			ΣΤ΄ Δημοτικού	YES	2,1451993	3	3	Απόκριες				Απόκριες	Απόκριες ονομάζονται	Γεωγραφί	1,810680475	0,003983255	1
994	ΑΠΟΡΡΙΜΜΑ	ΝΑΙ		Ε΄ Δημοτικού			7	6	Απορρίμματα	YES	14		Περιβάλλον,Ρύπανση	Ως απορρίμματα ή από	Γεωγραφί	1,516314976	0,002304215	1,166666667
995	ΑΠΟΡΡΙΠΤΩ			Ε΄ Δημοτικού	YES	2,3812222	1	1							Γεωγραφί	2,283301229	0,002994012	1
996	ΑΠΟΡΡΙΨΗ			Β΄ Γυμνασίου			2	2							Γεωλογία	1,984527313	0,001577632	1
997	ΑΠΟΡΡΟΗ			Α Γυμνασίου	YES	2,2245605	4	3							Γεωγραφί	1,810680475	0,004239479	1,333333333
998	ΑΠΟΡΡΟΙΑ			Α Γυμνασίου			1	1							Γεωλογία	2,283301229	0,001540832	1
999	ΑΠΟΡΡΟΦΩ	ΝΑΙ		Ε΄ Δημοτικού	YES	2,7590246	11	6							Γεωγραφί	1,516314976	0,005569162	1,833333333
1000	ΑΠΟΡΡΥΠΑΝΤΙΚΑ			Β΄ Γυμνασίου	YES	2,1257908	3	2	Απορρυπαντικό	YES	12		Προϊόντα καθαρισμοι	Ένα απορρυπαντικό είν	Γεωλογία	1,984527313	0,001783917	1,5
1001	ΑΠΟΣΑΘΡΩΣΗ	ΝΑΙ		ΣΤ΄ Δημοτικού	YES	3,327248	7	3	Αποσάθρωση				Γεωλογικά φαινόμενα	Με τον όρο αποσάθρωσ	Γεωγραφί	1,810680475	0,007223449	2,333333333
1002	ΑΠΟΣΠΑΣΜΑ			Ε΄ Δημοτικού			4	4	Αποσπάσματα α	YES	11		Στρατιωτική τακτική	Τα αποσπάσματα αποτ	Γεωγραφί	1,68797462	0,003352717	1
1003	ΑΠΟΣΠΩ			Ε΄ Δημοτικού			2	2							Γεωγραφί	1,984527313	0,002454879	1
1004	ΑΠΟΣΤΑΙΝΩ			Ε΄ Δημοτικού			7	7							Γεωγραφί	1,45156715	0,003538763	1
1005	ΑΠΟΣΤΑΣΗ	ΝΑΙ		Ε΄ Δημοτικού	YES	1,7716212	56	26							Γεωγραφί	0,921486386	0,006714205	2,153846154
1006	ΑΠΟΣΤΕΛΛΩ			ΣΤ΄ Δημοτικού			1	1							Γεωγραφί	2,283301229	0,002724796	1
1007	ΑΠΟΣΤΟΛΗ			Ε΄ Δημοτικού			2	2							Γεωγραφί	1,984527313	0,003409898	1
1008	ΑΠΟΣΤΟΛΙΑ			Α Γυμνασίου			1	1							Γεωλογία	2,283301229	0,003802281	1
1009	ΑΠΟΣΤΟΛΟΣ			Ε΄ Δημοτικού	YES	2,273447	1	1							Γεωγραφί	2,283301229	0,002770083	1
1010	ΑΠΟΣΤΡΑΓΓΙΖΩ	ΝΑΙ		Α Γυμνασίου	YES	2,6928093	1	1							Γεωλογία	2,283301229	0,004219409	1
1011	ΑΠΟΣΤΡΑΓΓΙΣΤΙΚΟΣ	ΝΑΙ		Β΄ Γυμνασίου	YES	2,5266364	2	1							Γεωλογία	2,283301229	0,002666667	2
1012	ΑΠΟΣΥΜΠΙΕΣΗ			Ε΄ Δημοτικού			1	1							Γεωγραφί	2,283301229	0,003448276	1
1013	ΑΠΟΣΥΝΘΕΣΗ			ΣΤ΄ Δημοτικού	YES	2,3373265	1	1							Γεωγραφί	2,283301229	0,003021148	1
1014	ΑΠΟΣΥΝΘΕΤΩ			Β΄ Γυμνασίου			1	1							Γεωλογία	2,283301229	0,001290323	1
1015	ΑΠΟΣΥΡΣΗ			Β΄ Γυμνασίου			1	1							Γεωλογία	2,283301229	0,000998004	1
1016	ΑΠΟΤΕΛΕΣΜΑ			Ε΄ Δημοτικού			151	85							Γεωγραφί	0,511490156	0,003738624	1,776470588
1017	ΑΠΟΤΕΛΕΣΜΑΤΙΚΑ			ΣΤ΄ Δημοτικού			3	2							Γεωγραφί	1,984527313	0,00246416	1,5
1018	ΑΠΟΤΕΛΕΣΜΑΤΙΚΟΣ			Ε΄ Δημοτικού			4	4							Γεωγραφί	1,68797462	0,002678312	1
1019	ΑΠΟΤΕΛΩ			Ε΄ Δημοτικού			346	118							Γεωγραφί	0,418076472	0,006174149	2,93220339
1020	ΑΠΟΤΙΜΩ			Β΄ Γυμνασίου			1	1							Γεωλογία	2,283301229	0,001175088	1
1021	ΑΠΟΤΟΜΑ			Ε΄ Δημοτικού	YES	1,7933811	7	6							Γεωγραφί	1,516314976	0,003489235	1,166666667
1022	ΑΠΟΤΟΜΟΣ	ΝΑΙ		Ε΄ Δημοτικού	YES	2,3855577	8	6							Γεωγραφί	1,516314976	0,003768739	1,333333333
1023	ΑΠΟΤΡΕΠΩ			Ε΄ Δημοτικού			2	2							Γεωγραφί	1,984527313	0,002070978	1
1024	ΑΠΟΤΥΠΩΜΑ			Α Γυμνασίου	YES	2,309257	2	2	Αποτύπωμα				CS1 maint: Πολλαπλές	Το αποτύπωμα είναι σημ	Γεωλογία	1,984527313	0,002685978	1

Προτάσεις με επιλεγμένη λέξη και συνώνυμα / αντίθετα / λάθη

Το συγκεκριμένο αρχείο (excel - ebooks_lexicon_dimitikou_a_b_c_sample_sentences_antonyms_incorrect_answers_v1.xlsx) περιέχει προτάσεις από το Λεξικό Α-Β-Γ του δημοτικού, στις οποίες έχει επιλεγεί και αφαιρεθεί μια λέξη. Για τη λέξη αυτή, δίδεται το λήμμα, ενώ έχουν παραχθεί και συνώνυμα / αντώνυμα για αξιοποίηση στα παιχνίδια.

Λεξιπαίγνιο: Π3.2 Αναφορά Τεκμηρίωσης



	A	B	C	D	E
1	lemma	text	correct_answer	incorrect_answers	strong_incorrect_answers
2	άβολος	«Αυτή η καρέκλα είναι μικρή και ... για το θείο Αλέκο» είπε ο Κώστας. Ο θείος έχει μεγάλα πόδια και δε βολεύεται.	άβολη	εξυπηρετική, άνετη, βολική	βολική, άνετη,
3	άδειος	Ο Κώστας άνοιξε το κουτί για να πάρει μία σοκολάτα αλλά δε βρήκε τίποτα. Ήταν ...	άδειο	παχουλό, αφρατούλικο, γεμάτο	γεμάτο,
4	άξιος	Η κυρία Μαργαρίτα είναι ... μαγείρισσα.	άξια	ασήμαντα, τιποτένια, ανάξια	ανάξια,
5	άπλυτος	«Κώστα, τι ... χέρια είναι αυτά! Πήγαινε να τα πλύνεις γρήγορα!» είπε η κυρία Μαργαρίτα.	άπλυτα	διαυγή, καθαρά, ξεκάθαρα	καθαρά,
6	άσχημος	Η κακιά βασίλισσα ήταν ... και ζήλευε τη Χιονάτη, που ήταν όμορφη.	άσχημη	ωραία, όμορφη, νόστιμη	όμορφη, ωραία,
7	άτυχος	Ο Κώστας ήταν πολύ ... χθες. Έπεσε και χτύπησε το πόδι του στο ποδόσφαιρο.	άτυχος	καλότυχος, αίσιος, τυχερός	τυχερός,
8	αδειάζω	Η Χιονάτη ... το σκουπιδοτενεκέ που ήταν γεμάτος χαρτιά.	άδειασε	γέμισε, κατάκλυσε, γόμωσε	γέμισε,
9	αδύνατος	Ο Κώστας είναι ... , ενώ ο θείος Αλέκος ψηλός και χοντρός.	αδύνατος	γεμάτος, χοντρός, παχύς	παχύς, χοντρός,
10	ανήσυχος	Ο Κώστας είχε ... ύπνο, γιατί τον ενοχλούσε το δόντι του όλο το βράδυ.	ανήσυχο	ήσυχο, πράο, ήρεμο	ήσυχο, ήρεμο,
11	ανίκανος	Η Αλίκη ήταν τόσο κουρασμένη που ήταν ... να κάνει έστω κι ένα βήμα παραπάνω.	ανίκανη	επιδέξια, άξια, ικανή	ικανή, επιδέξια,
12	αναίσθητος	Στο έργο που είδε η θεία Κατερίνα ο αστυνομικός έριξε μία μπουνιά στον κλέφτη και τον έριξε κάτω	αναίσθητο	ευαίσθητο, ευάλωτο, ευπαθές	ευαίσθητο,
13	ανεβάζω	Οι εργάτες ... τα έπιπλα από το ισόγειο στον τρίτο όροφο του σπιτιού.	ανέβασαν	κατέβασαν, μείωσαν, καταβρόχθισαν	κατέβασαν,
14	ανεβαίνω	Στην εκδρομή ο Κώστας και οι φίλοι του ήθελαν να ... στην κορυφή του βουνού.	ανεβούν	κατεβούν, κατέβουν, ξεκαβαλικέψουν	κατεβούν, κατέβουν,
15	ανησυχώ	Η Αθηνά ... , γιατί η Ροζαλία δε γύρισε ακόμη.	ανησυχεί	ξενοιάζει, ησυχάζει, ηρεμεί	ησυχάζει, ηρεμεί,
16	ανυπάκουος	Ο Νίκος είναι ... μαθητής και γι' αυτό δέχεται συχνά τιμωρίες.	ανυπάκουος	πειθαρχικός, πειθήνιος, υπάκουος	υπάκουος, πειθαρχικός,
17	ανυπόφορος	Στο μαγαζί του κυρίου Δημήτρη είχε σήμερα ... ζέστη, κι έτσι εκείνος άνοιξε όλα τα παράθυρα.	ανυπόφορη	καλούτσικη, ανεκτή, υποφερτή	υποφερτή,
18	ανόητος	«Τι ... που είναι ο Νίκος» σκέφτηκε η Αθηνά. «Κυνηγάει τη Ροζαλία που δεν τον πείραξε ποτέ σε τίποτα».	ανόητος	εύστροφος, πονηρός, έξυπνος	έξυπνος,
19	απλώνω	Μετά το φαγητό η κυρία Μαργαρίτα άρχισε ν' ... τα ρούχα στο μπαλκόνι.	απλώνει	ελκύει, σηκώνει, μαζεύει	μαζεύει,
20	απολύω	Η εταιρεία που είναι δίπλα στο γραφείο του Γιάννη ... πέντε υπαλλήλους.	απέλυσε	ανέλαβε, διόρισε, προσέλαβε	προσέλαβε,
21	αποτυχία	Παρά την προσπάθεια που έκανε η Αθηνά πήρε κακό βαθμό στην ορθογραφία. Η ... της τη στενοχώρησε πολύ.	αποτυχία	επιτυχία, επίτευξη, ευστοχία	επιτυχία,
22	απουσία	«Ο Νίκος κάνει πολλές ... στο σχολείο τελευταία. Μήπως είναι άρρωστος;» ρώτησε η δασκάλα τον Κώστα.	απουσίες	παρουσίες, προσελεύσεις, υπάρξεις	παρουσίες,
23	αργά	Η Αθηνά τώρα μαθαίνει ποδήλατο, γι' αυτό δεν τρέχει, πηγαίνει	αργά	βιαστικά, ταχέως, γρήγορα	γρήγορα, βιαστικά,
24	αρνούμαι	Ο Κώστας ... το δεύτερο γλυκό, γιατί πονούσε η κοιλιά του.	αρνήθηκε	υποδέχτηκε, θεωρήθηκε, δέχτηκε	δέχτηκε,
25	αρχίζω	Ο Κώστας ... να παίζει ποδόσφαιρο, όταν ήταν τεσσάρων χρονών.	άρχισε	τέλειωσε, τελείωσε, τερμάτισε	τελείωσε, τέλειωσε,
26	ασήμαντος	Πολλοί νόμιζαν ότι η Βούλα Πατουλίδου ήταν ... αθλήτρια, μέχρι που κέρδισε το χρυσό μετάλλιο στους Ολυμπιακούς αγώνες.	ασήμαντη	σημαντική, μεγάλη, σπουδαία	σημαντική, μεγάλη, σπουδαία
27	ασφάλεια	Από τότε που ο κύριος Δημήτρης έβαλε κάγκελα στα παράθυρα του μαγαζιού του, αισθάνεται μεγαλύτερη	ασφάλεια	διακινδύνευση, ανασφάλεια, απειλή	ανασφάλεια,
28	αυξάνω	Με την καινούρια χρονιά οι γονείς του Κώστα ... το χαρτζιλίκι του κατά 50 λεπτά.	αύξησαν	λιγόστεψαν, ελάττωσαν, μείωσαν	ελάττωσαν,
29	αφήνω	Ο Κώστας ... την τσάντα του στο δωμάτιό του και πήγε στην κουζίνα να φάει.	άφησε	πήρε, πάρε, πιάσε	πάρε, πήρε,
30	αόρατος	«Τα μικρόβια είναι ... χωρίς μικροσκόπιο» είπε ο γιατρός στον Κώστα.	αόρατα	θεατά, ορατά, εμφανή	ορατά,
31	βγάζω	Ο θείος Τάκης είχε ταξίδι. ... τα ρούχα του από την ντουλάπα και τα έβαλε στη βαλίτσα του.	έβγαλε	ενέταξε, σημείωσε, έβαλε	έβαλε,
32	γίγαντας	Οι παίκτες του μπάσκετ είναι πολύ ψηλοί, μοιάζουν με	γίγαντες	νάνους, ζουμπάδες, νάνοι	νάνους, νάνοι,
33	γεμίζω	Ο Κώστας ... τις τσέπες του με καραμέλες και βγήκε έξω να τις φάει με την Αθηνά.	γέμισε	έριξε, ρίξε, άδειασε	άδειασε,
34	γενναίος	Ο Άγκλι είναι πολύ ... παιδί. Πήγε μόνος του στον κύριο Μιχάλη και του είπε ότι αυτός έσπασε κατά λάθος το τζάμι του.	γενναίο	συνεσταλμένο, δειλό, άτολμο	δειλό,
35	γκρεμίζω	Το κύμα ... το κάστρο που έχτισαν η Αθηνά και ο Κώστας στην άμμο.	γκρέμισε	χτίσε, έφτιαξε, έχτισε,	έχτισε, χτίσε, έφτιαξε, φτιάξε
36	δέχομαι	Ο θείος Σταμάτης ... πολλά δώρα στη γιορτή του.	δέχτηκε	αφέθηκε, αρνήθηκε, αποκηρύχτηκε	αρνήθηκε,
37	δειλός	Ο Κώστας είναι ... και φοβάται να πάει στον οδοντίατρο.	δειλός	θαρραλέος, γενναίος, ανδρείος	θαρραλέος, γενναίος,
38	δημόσιος	Στη γειτονιά της Αθηνάς υπάρχει μία ... βιβλιοθήκη για όλους τους Αθηναίους.	δημόσια	προσωπική, ιδιωτικά, ιδιωτική	ιδιωτική, ιδιωτικά,
39	διαιρώ	«Αν ... το οκτώ με το δύο, παίρνουμε τέσσερα» είπε η δασκάλα.	διαιρέσουμε	πολλαπλασιάσουμε, πληθύνουμε, αυξήσουμε	πολλαπλασιάσουμε,
40	διαφέρω	Η κασετίνα της Αθηνάς ... από εκείνη της Ελένης: η μία είναι ξύλινη και καφέ και η άλλη πλαστική και κόκκινη.	διαφέρει	μοιάσει, δείξει, μοιάζει	μοιάσει, μοιάζει,
41	διαφωνία	Μετά τη ... τους για το ποιος θα γίνει αρχηγός στο παιχνίδι ο Κώστας και ο Νίκος ξανάγιναν φίλοι.	διαφωνία	συνεννόηση, συμφωνία, αντιστοιχία	συμφωνία,
42	δυναμώνω	Η γυμναστική ... το σώμα. Γι' αυτό και η Αλίκη τρέχει κάθε μέρα.	δυναμώνει	ελαττώνει, κατεβάζει, χαμηλώνει	χαμηλώνει,
43	δυσάρεστος	Τα νέα ήταν πολύ ... για τον Κώστα. Ο καιρός χάλασε και η εκδρομή δε μπορούσε να γίνει.	δυσάρεστα	χαρούμενα, ευάρεστα, ευχάριστα	ευχάριστα,

default_1

Ready

Λέξεις με προθήματα

Το συγκεκριμένο αρχείο (excel - preposition_lexicon_lemma_v1.xlsx) περιέχει λέξεις με προθήματα από το μορφολογικό λεξικό, με μια σειρά από πληροφορίες στις επόμενες στήλες σε σχέση με τη γραμματική κατηγορία της λέξης, σχετικές λέξεις, σχετικές φράσεις, κτλ.

F90 -ος, -η, -ο επίρρ. -α

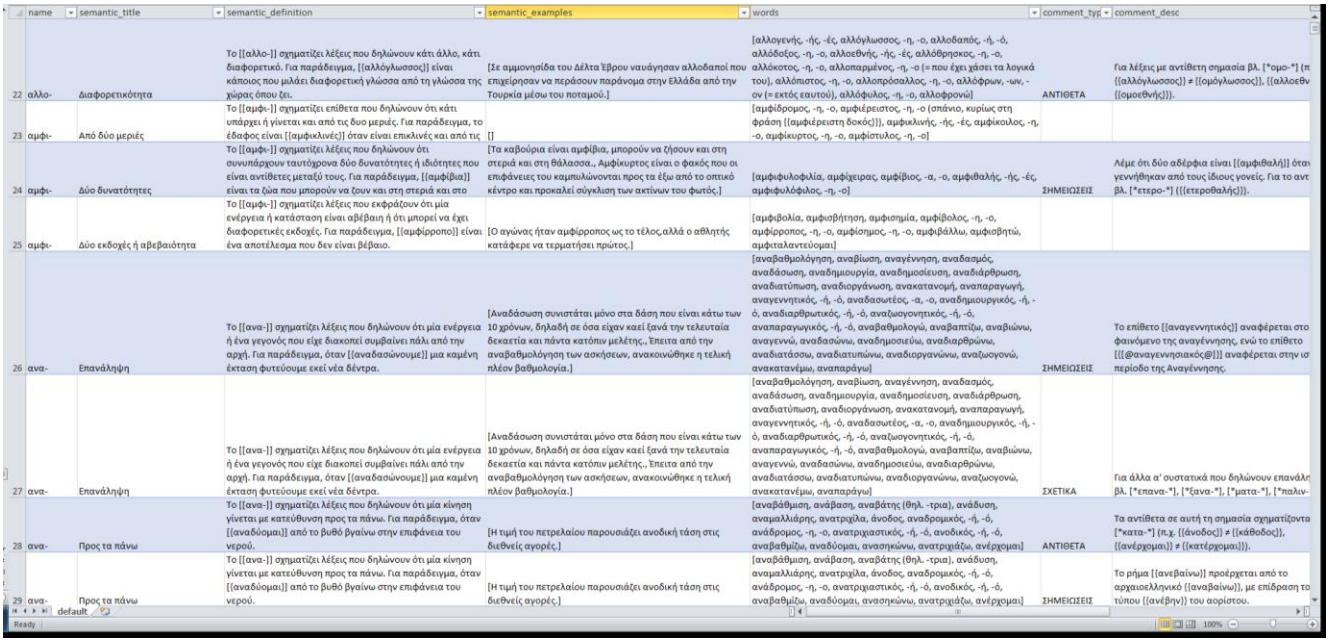
	NORMA	prepositi	hw	pos	description	name	RELWORDS	RELPHRASES	INFORMAL	KSENI	ANCIENT
172	A	ά	άθολος	ADJ	-ος, -η, -ο όχι επίρρ. = όχι θολός	άθολος					
173	A	ά	άθραυστος	ADJ	-ος, -η, -ο όχι επίρρ.	άθραυστος					
174	A	ά	άθρεφτος	ADJ	-ος, -η, -ο όχι επίρρ.	άθρεφτος					
175	A	ά	άθρησκα	ADV		άθρησκα					
176	A	ά	άθρησκος	ADJ	-ος, -η, -ο επίρρ. -α	άθρησκος					
177	A	ά	άθρονος	ADJ	-ος, -η, -ο όχι επίρρ.	άθρονος					
178	A	ά	άθρυπτος	ADJ	-ος, -η, -ο όχι επίρρ. Μ λήμμα	άθρυπτος	άθρυφτος				
179	A	ά	άθρυφτος	ADJ	-ος, -η, -ο όχι επίρρ. Μ λήμμα	άθρυφτος	άθρυπτος		άθρυπτος		
180	A	ά	άθυμα	ADV		άθυμα					
181	A	α	άθυμα	ADV		άθυμα					
182	A	ά	άθυμος	ADJ	-ος, -η, -ο επίρρ. -α, -ως λόγ. Κ, Μ μόνο λήμμα	άθυμος					άκεφος
183	A	ά	άθυρμα	N	λόγ. Μ, Τρ μόνο λήμμα	άθυρμα					μαριονέτα
184	A	α	άθυρμα	N	λόγ. Μ, Τρ μόνο λήμμα	άθυρμα					μαριονέτα
185	A	ά	άθυρος	ADJ	-ος, -η, -ο όχι επίρρ.	άθυρος					
186	A	ά	άκαγος	ADJ	-ος, -η, -ο όχι επίρρ. Τρ μόνο λήμμα	άκαγος	άκαιγος,άκαος				
187	A	ά	άκαιγος	ADJ	-ος, -η, -ο όχι επίρρ.	άκαιγος	άκαος,άκαγος				
188	A	ά	άκαιρα	ADV		άκαιρα		ευκαίρως ακαίρως			
189	A	α	άκαιρα	ADV		άκαιρα		ευκαίρως ακαίρως			
190	A	ά	άκαιρο	N		άκαιρο	άκαιρος				
191	A	ά	άκαιρος	ADJ	-ος, -η, -ο επίρρ. -α, -ως	άκαιρος	άκαιρο,ακαίρως				
192	A	ά	άκακα	ADV		άκακα					
193	A	α	άκακα	ADV		άκακα					
194	A	ά	άκακος	ADJ	-ος, -η, -ο επίρρ. -α	άκακος		είναι άκακο αρνί			
195	A	ά	άκαμπτα	ADV		άκαμπτα					
196	A	ά	άκαμπτος	ADJ	-ος, -η, -ο επίρρ. -α	άκαμπτος					
197	A	ά	άκαος	ADJ	-ος, -η, -ο όχι επίρρ.	άκαος	άκαγος,άκαιγος				
198	A	ά	άκαπνος	ADJ	-ος, -η, -ο επίρρ. -α οικ. = χωρίς τσιγάρα,μειωτ., ειρων. = απόλεμος	άκαπνος,Τρ.					
199	A	ά	άκαρδα	ADV		άκαρδα					
200	A	ά	άκαρδος	ADJ	-ος, -η, -ο επίρρ. -α	άκαρδος					
201	A	ά	άκαρι	N	Μ, Μπ μόνο λήμμα -εως, -εα, -εων	άκαρι		ψωρικό άκαρι			
202	A	α	άκαρι	N	Μ, Μπ μόνο λήμμα -εως, -εα, -εων	άκαρι		ψωρικό άκαρι			
203	A	ά	άκαρπα	ADV		άκαρπα					
204	A	ά	άκαρπος	ADJ	-ος, -η, -ο επίρρ. -α άτεκνος όλοι λήμμα	άκαρπος					
205	A	ά	άκαυστα	ADV		άκαυστα					άκαυτα
206	A	ά	άκαυστος	ADJ	-ος, -η, -ο επίρρ. -α λόγ	άκαυστος	άκαυτος,ακαής				άκαυτος
207	A	ά	άκαυτος	ADJ	-ος, -η, -ο επίρρ. -α;;; οικ. όλοι λήμμα	άκαυτος	ακαής		άκαυστος		
208	A	ά	άκεντρος	ADJ	-ος, -η, -ο όχι επίρρ. Μ μόνο λήμμα	άκεντρος					
209	A	ά	άκερδα	ADV		άκερδα					
210	A	ά	άκερδος	ADJ	-ος, -η, -ο επίρρ. -α	άκερδος	ακερδής				
211	A	ά	άκερκος	ADJ	-ος, -η, -ο όχι επίρρ. Μ λήμμα	άκερκος					
212	A	ά	άκερος	ADJ	-ος, -η, -ο όχι επίρρ.	άκερος	ακέρατος				
213	A	ά	άκεφα	ADV		άκεφα					
214	A	ά	άκεφος	ADJ	-ος, -η,-ο επίρρ.-α	άκεφος					

default

Ready

Σημασία των προθημάτων στις λέξεις

Το συγκεκριμένο αρχείο (excel - xtizo_lekseis_prefixes_v1.xlsx) περιέχει το πρώτο συστατικό σύνθετων λέξεων, με τη σημασία του, παραδείγματα χρήσεις αλλά και παραδείγματα λέξεων που ξεκινούν με αυτό. Το υλικό προέρχεται κυρίως από το «Χτίζω Λέξεις».



	name	semantic_title	semantic_definition	semantic_examples	words	comment_type	comment_desc
22	αλλο-	Διαφορετικότητα	Το [[αλλο-]] σχηματίζει λέξεις που δηλώνουν κάτι άλλο, κάτι διαφορετικό. Για παράδειγμα, [[αλλόγλωσσος]] είναι κάποιος που μιλάει διαφορετική γλώσσα από τη γλώσσα της χώρας όπου ζει.	[Σε αμμονησίδα του Δέλτα Έβρου ναυάγησαν αλλοδαποί που επιχείρησαν να περάσουν παράνομα στην Ελλάδα από την Τουρκία μέσω του ποταμού.]	[αλλογενής, -ής, -ές, αλλόγλωσσος, -η, -ο, αλλοδαπός, -ή, -ό, αλλόδοξος, -η, -ο, αλλοεθνής, -ής, -ές, αλλόθρησκος, -η, -ο, αλλόκοτος, -η, -ο, αλλοπαρμένος, -η, -ο (= που έχει χάσει τα λογικά του), αλλόπιστος, -η, -ο, αλλοπρόσαλλος, -η, -ο, αλλόφρων, -ων, -ον (= εκτός εαυτού), αλλόφυλος, -η, -ο, αλλοφρονώ]	ΑΝΤΙΘΕΤΑ	Για λέξεις με αντίθετη σημασία βλ. [*ομο-*] (π [[αλλόγλωσσος]] ≠ [[ομόγλωσσος]], [[αλλοεθν [[ομοεθνής]]).
23	αμφι-	Από δύο μεριές	Το [[αμφι-]] σχηματίζει επίθετα που δηλώνουν ότι κάτι υπάρχει ή γίνεται και από τις δυο μεριές. Για παράδειγμα, το έδαφος είναι [[αμφικλινές]] όταν είναι επικλινές και από τις	[]	[αμφίδρομος, -η, -ο, αμφιέρειστος, -η, -ο (σπάνιο, κυρίως στη φράση {{αμφιέρειστη δοκός}}), αμφικλινής, -ής, -ές, αμφίκοιλος, -η, -ο, αμφίκυρτος, -η, -ο, αμφίστυλος, -η, -ο]		
24	αμφι-	Δύο δυνατότητες	Το [[αμφι-]] σχηματίζει λέξεις που δηλώνουν ότι συνυπάρχουν ταυτόχρονα δύο δυνατότητες ή ιδιότητες που είναι αντίθετες μεταξύ τους. Για παράδειγμα, [[αμφίβια]] είναι τα ζώα που μπορούν να ζουν και στη στεριά και στο	[Τα καβούρια είναι αμφίβια, μπορούν να ζήσουν και στη στεριά και στη θάλασσα., Αμφίκυρτος είναι ο φακός που οι επιφάνειες του καμπυλώνονται προς τα έξω από το οπτικό κέντρο και προκαλεί σύγκλιση των ακτίνων του φωτός.]	[αμφιφυλοφιλία, αμφίχειρας, αμφίβιος, -α, -ο, αμφιθαλής, -ής, -ές, αμφιφυλόφιλος, -η, -ο]	ΣΗΜΕΙΩΣΕΙΣ	Λέμε ότι δύο αδέρφια είναι [[αμφιθαλή]] όταν γεννήθηκαν από τους ίδιους γονείς. Για το αντ βλ. [*ετερο-*] ({{ετεροθαλής}}).
25	αμφι-	Δύο εκδοχές ή αβεβαιότητα	Το [[αμφι-]] σχηματίζει λέξεις που εκφράζουν ότι μία ενέργεια ή κατάσταση είναι αβέβαιη ή ότι μπορεί να έχει διαφορετικές εκδοχές. Για παράδειγμα, [[αμφίρροπο]] είναι ένα αποτέλεσμα που δεν είναι βέβαιο.	[Ο αγώνας ήταν αμφίρροπος ως το τέλος,αλλά ο αθλητής κατάφερε να τερματήσει πρώτος.]	[αμφιβολία, αμφισβήτηση, αμφισημία, αμφίβολος, -η, -ο, αμφίρροπος, -η, -ο, αμφίσημος, -η, -ο, αμφιβάλλω, αμφισβητώ, αμφιταλαντεύομαι]		
26	ανα-	Επανάληψη	Το [[ανα-]] σχηματίζει λέξεις που δηλώνουν ότι μία ενέργεια ή ένα γεγονός που είχε διακοπεί συμβαίνει πάλι από την αρχή. Για παράδειγμα, όταν [[αναδασώνουμε]] μια καμένη έκταση φυτεύουμε εκεί νέα δέντρα.	[Αναδάσωση συνιστάται μόνο στα δάση που είναι κάτω των 10 χρόνων, δηλαδή σε όσα είχαν καεί ξανά την τελευταία δεκαετία και πάντα κατόπιν μελέτης., Έπειτα από την αναβαθμολόγηση των ασκήσεων, ανακοινώθηκε η τελική πλέον βαθμολογία.]	[αναβαθμολόγηση, αναβίωση, αναγέννηση, αναδασμός, αναδάσωση, αναδημιουργία, αναδημοσίευση, αναδιάρθρωση, αναδιατύπωση, αναδιοργάνωση, ανακατανομή, αναπαραγωγή, αναγεννητικός, -ή, -ό, αναδασωτέος, -α, -ο, αναδημιουργικός, -ή, -ό, αναδιαρθρωτικός, -ή, -ό, αναζωογονητικός, -ή, -ό, αναπαραγωγικός, -ή, -ό, αναβαθμολογώ, αναβαπτίζω, αναβιώνω, αναγεννώ, αναδασώνω, αναδημοσιεύω, αναδιαρθρώνω, αναδιατάσσω, αναδιατυπώνω, αναδιοργανώνω, αναζωογονώ, ανακατανέμω, αναπαράγω]	ΣΗΜΕΙΩΣΕΙΣ	Το επίθετο [[αναγεννητικός]] αναφέρεται στο φαινόμενο της αναγέννησης, ενώ το επίθετο [[[@αναγεννησιακός@]]] αναφέρεται στην ισ περίοδο της Αναγέννησης.
27	ανα-	Επανάληψη	Το [[ανα-]] σχηματίζει λέξεις που δηλώνουν ότι μία ενέργεια ή ένα γεγονός που είχε διακοπεί συμβαίνει πάλι από την αρχή. Για παράδειγμα, όταν [[αναδασώνουμε]] μια καμένη έκταση φυτεύουμε εκεί νέα δέντρα.	[Αναδάσωση συνιστάται μόνο στα δάση που είναι κάτω των 10 χρόνων, δηλαδή σε όσα είχαν καεί ξανά την τελευταία δεκαετία και πάντα κατόπιν μελέτης., Έπειτα από την αναβαθμολόγηση των ασκήσεων, ανακοινώθηκε η τελική πλέον βαθμολογία.]	[αναβαθμολόγηση, αναβίωση, αναγέννηση, αναδασμός, αναδάσωση, αναδημιουργία, αναδημοσίευση, αναδιάρθρωση, αναδιατύπωση, αναδιοργάνωση, ανακατανομή, αναπαραγωγή, αναγεννητικός, -ή, -ό, αναδασωτέος, -α, -ο, αναδημιουργικός, -ή, -ό, αναδιαρθρωτικός, -ή, -ό, αναζωογονητικός, -ή, -ό, αναπαραγωγικός, -ή, -ό, αναβαθμολογώ, αναβαπτίζω, αναβιώνω, αναγεννώ, αναδασώνω, αναδημοσιεύω, αναδιαρθρώνω, αναδιατάσσω, αναδιατυπώνω, αναδιοργανώνω, αναζωογονώ, ανακατανέμω, αναπαράγω]	ΣΧΕΤΙΚΑ	Για άλλα α' συστατικά που δηλώνουν επανάλη βλ. [*επανα-*], [*ξανα-*], [*ματα-*], [*παλιν-
28	ανα-	Προς τα πάνω	Το [[ανα-]] σχηματίζει λέξεις που δηλώνουν ότι μία κίνηση γίνεται με κατεύθυνση προς τα πάνω. Για παράδειγμα, όταν [[αναδύομαι]] από το βυθό βγαίνω στην επιφάνεια του νερού.	[Η τιμή του πετρελαίου παρουσιάζει ανοδική τάση στις διεθνείς αγορές.]	[αναβάθμιση, ανάβαση, αναβάτης (θηλ. -τρια), ανάδυση, αναμαλλιάρης, ανατρχίλα, άνοδος, αναδρομικός, -ή, -ό, ανάδρομος, -η, -ο, ανατριχαστικός, -ή, -ό, ανοδικός, -ή, -ό, αναβαθμίζω, αναδύομαι, ανασηκώνω, ανατριχιάζω, ανέρχομαι]	ΑΝΤΙΘΕΤΑ	Τα αντίθετα σε αυτή τη σημασία σχηματίζοντα [*κατα-*] (π.χ. [{άνοδος}] ≠ [{κάθοδος}], [{ανέρχομαι}] ≠ [{κατέρχομαι}]).
29	ανα-	Προς τα πάνω	Το [[ανα-]] σχηματίζει λέξεις που δηλώνουν ότι μία κίνηση γίνεται με κατεύθυνση προς τα πάνω. Για παράδειγμα, όταν [[αναδύομαι]] από το βυθό βγαίνω στην επιφάνεια του νερού.	[Η τιμή του πετρελαίου παρουσιάζει ανοδική τάση στις διεθνείς αγορές.]	[αναβάθμιση, ανάβαση, αναβάτης (θηλ. -τρια), ανάδυση, αναμαλλιάρης, ανατριχίλα, άνοδος, αναδρομικός, -ή, -ό, ανάδρομος, -η, -ο, ανατριχαστικός, -ή, -ό, ανοδικός, -ή, -ό, αναβαθμίζω, αναδύομαι, ανασηκώνω, ανατριχιάζω, ανέρχομαι]	ΣΗΜΕΙΩΣΕΙΣ	Το ρήμα [[ανεβαίνω]] προέρχεται από το αρχαιοελληνικό [[αναβαίνω]], με επίδραση το τύπου [[ανέβην]] του αορίστου.

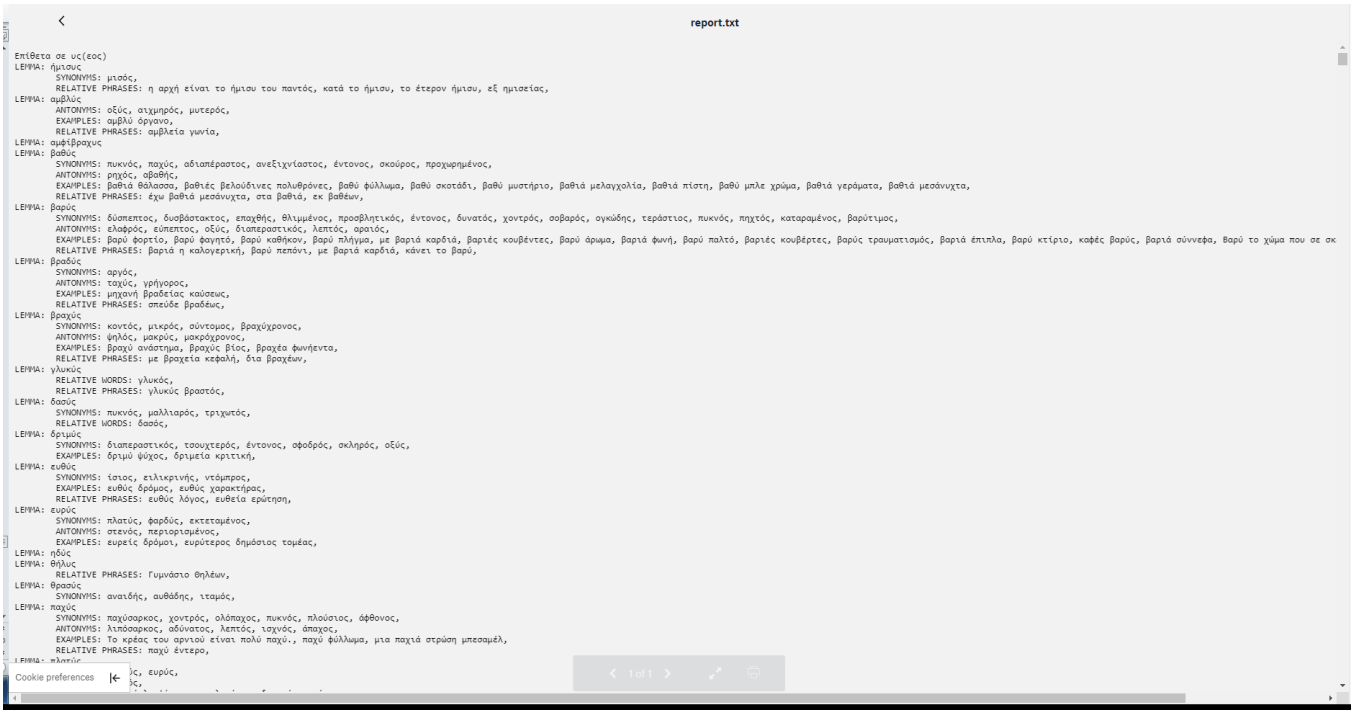
default

Ready

100%

Επίθετα τριγενή και δικατάληκτα

Το συγκεκριμένο αρχείο (txt - report.txt) περιέχει πληροφορίες για επίθετα τριγενή και δικατάλικτα, προερχόμενες κυρίως από το Θησαυρό.



Αποτελέσματα στατιστικής επεξεργασίας δεδομένων (Αρχεία)

Πέρα από την επεξεργασία με μεθόδους ΕΦΓ (NLP) που έγιναν και τα αποτελέσματα περιγράφονται στα παραπάνω, όπως περιγράφηκε και στο κεφ. 2.2, έγινε και στατιστική επεξεργασία των δεδομένων (και κυρίως των περιεχομένων των σχολικών βιβλίων). Κάποια αποτελέσματα σε σχέση με αυτά δίνονται στο Π3.3 και καταγράφονται στο Σχετικό παράρτημα (1.2).

Έτσι στο παρακάτω παράδειγμα, βλέπουμε το αποτέλεσμα εύρεσης πολυλεκτικών όρων σε βιβλίο της γεωγραφίας, όπου κατασημαίνεται η συχνότητα εμφάνισης του πρώτου όρου, του δεύτερου, αλλά και του συνδυασμού τους.

```
αδιάκοπη/2 κίνηση/1     1
αδιάκοπη/2 ανάπτυξη/1   1
δυτικότερο/2 σημείο/2   2
ποικιλόμορφο/1 ανάγλυφο/1     1
ξακουστός/1 βασιλιάς/1  1
τρίτο/3 επίπεδο/1 1
τρίτο/3 θρανίο/1  1
τρίτο/3 άνοιγμα/1 1
κίτρινη/1 φυλή/1  1
αρχαιολογικές/1 ανασκαφές/1   1
ευρύτερης/1 περιοχής/1  1
αέριες/1 μάζες/1  1
τρίτη/14 σειρά/1  1
τρίτη/14 οροσειρά/1     1
τρίτη/14 ομάδα/11 11
τρίτη/14 στήλη/1  1
θρησκευτικούς/1 λόγους/1      1
ανατολικής/10 Ευρώπης/4 4
ανατολικής/10 Αφρικής/1 1
ανατολικής/10 Ελλάδας/3 3
ανατολικής/10 Αμερικής/1      1
ανατολικής/10 Φινλανδίας/1    1
εμφύλιος/1 πόλεμος/1    1
εμφύλιοι/1 πόλεμοι/1    1
χορτοφάγα/1 ζώα/1 1
πολικού/1 κλίματος/1    1
Συχνό/1 φαινόμενο/1     1
Ευρωπαϊκής/57 Επιτροπής/5     5
Ευρωπαϊκής/57 Ένωσης/52 52
ραγδαία/4 αύξηση/1      1
ραγδαία/4 ανάπτυξη/1    1
ραγδαία/4 ανάπτυξη/2    2
οδικές/1 μεταφορές/1    1
φιλόξενη/1 θάλασσα/1    1
θεμελιωδών/1 Δικαιωμάτων/1    1
Βαλτική/8 θάλασσα/8     8
Πολλά/7 κράτη/1   1
Πολλά/7 νησιά/1   1
Πολλά/7 στοιχεία/1      1
Πολλά/7 προϊόντα/1      1
Πολλά/7 χρόνια/1  1
Πολλά/7 ποτάμια/2 2
Οδική/1 αρτηρία/1 1
ανατολικές/2 ακτές/2    2
υψηλού/1 επιπέδου/1     1
φορολογικό/1 σύστημα/1  1
βασικοί/1 παράγοντες/1  1
συνεταιριστικό/1 επίπεδο/1    1
μαζικό/1 τουρισμό/1     1
σεισμολογικοί/1 σταθμοί/1     1
ιαματικών/1 πηγών/1     1
υπόλοιπους/2 πλανήτες/1 1
υπόλοιπους/2 λαούς/1    1
ευαίσθητες/1 περιοχές/1 1
Συχνά/1 σ/1 1
μελλοντικής/1 άμμου/1   1
γόνιμο/2 έδαφος/2 2
αρχαίας/3 Ελλάδας/2     2
αρχαίας/3 γλώσσας/1     1
γόνιμη/1 κοιλάδα/1      1
μυθικού/1 Οδυσσέα/1     1
ολλανδική/1 πόλη/1      1
χημικών/2 στοιχείων/1   1
χημικών/2 προϊόντων/1   1
γόνιμα/2 εδάφη/2  2
παγωμένες/2 εκτάσεις/1  1
παγωμένες/2 έρημοι/1    1
αστικά/2 κέντρα/2 2
Διάσημοι/1 άνθρωποι/1   1
μελλοντικές/1 ενέργειές/1     1
ακόλουθο/4 σκίτσο/1     1
ακόλουθο/4 πίνακα/2     2
ακόλουθο/4 χάρτη/1      1
κυριαρχα/1 έθνη/1 1
Ιόνιο/7 πέλαγος/5 5
Ιόνιο/7 Πέλαγος/2 2
```

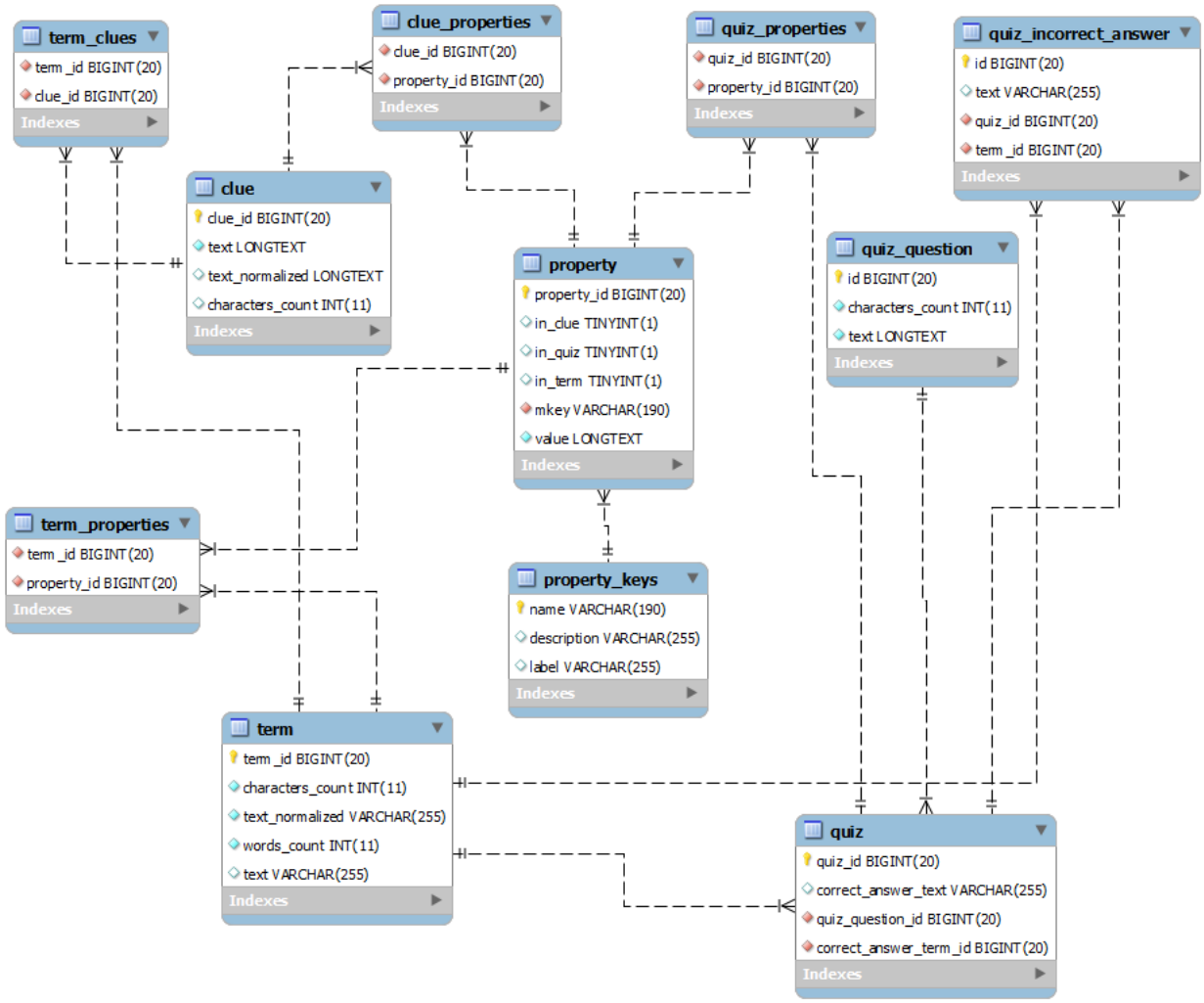
Βάση Δεδομένων

Στα πλαίσια του έργου αναπτύχθηκε μια βάση δεδομένων για την αποθήκευση του χαρακτηρισμένου γλωσσικό υλικό. Η αποθήκευση του υλικού στη βάση δεδομένων το κάνει αυτόματα προσπελάσιμο στους χρήστες μέσω της πληθώρας εργαλείων που υπάρχουν στη αγορά. Η υλοποίηση της βάσης έγινε στην MySQL και περιέχει το σύνολο του υλικού (δεδομένων και μεταδεδομένων) που χρησιμοποιείται από τα παιχνίδια με εξαίρεση το υλικό των binary λεξικών του έργου. Το υλικό στα binary λεξικά είναι προσπελάσιμα μέσω των services της πλατφόρμας Mnemosyne.

Οι κύριοι πίνακες της βάσης δεδομένων όπου είναι αποθηκευμένη η πληροφορία που χρησιμοποιούν τα παιχνίδια είναι οι εξής:

- Πίνακας "**term**": Στο πίνακα αυτό αποθηκεύονται όλα τα μεμονωμένα λεκτικά (term - όροι) που χρησιμοποιούνται στα παιχνίδια. Αυτά μπορεί να είναι μονολεκτικά ή πολυλεκτικά. Π.χ. εδώ αποθηκεύονται οι λέξεις για το παιχνίδι της κρεμάλας ή οι επιλογές ενός κουίζ.
- Πίνακας "**clue**": Εδώ αποθηκεύονται προτάσεις που μπορεί να είναι ορισμοί, ερμηνεία, γρίφοι, ενδείξεις, γνωμικά κ.α. και συνδέονται με κάποια λεκτικά(terms - όρους).
- Πίνακας "**term_clues**": Συνδέει τα λεκτικά(terms) με τις προτάσεις(clues)
- Πίνακας "**quiz**": Εδώ αποθηκεύονται η πληροφορία για το ποια είναι η ερώτηση και ποια η σωστή απάντηση σε ένα κουίζ.
- Πίνακας "**quiz_question**": Εδώ αποθηκεύονται οι ερωτήσεις των κουίζ.
- Πίνακας "**quiz_incorrect_answer**": Εδώ αποθηκεύονται οι επιλογές λανθασμένων απαντήσεων (terms) στα κουίζ.
- Πίνακας "**property**": Εδώ αποθηκεύεται τα μεταδεδομένα και η πληροφορία για τον χαρακτηρισμό του υλικού. Κάθε χαρακτηριστικό(property) έχει ένα όνομα και μια τιμή.
- Πίνακας "**term_properties**", "**clue_properties**" και "**quiz_properties**": Είναι οι πίνακες που συνδέουν τα χαρακτηριστικά με τα λεκτικά, τις προτάσεις και τα κουίζ.

Η παρακάτω εικόνα δείχνει το σχήμα της βάσης.



Java EE Web RESTful API

Για την ανάγκη, τόσο των παιχνιδιών, όσο και των απλών χρηστών, για να υπάρχει πρόσβαση στο χαρακτηρισμένο υλικό του έργου μέσω του web έχουν υλοποιηθεί μια σειρά από RESTful Web Services σε τεχνολογίες Java. Αυτά τα Web Services συνθέτουν το Web API που έχει αναπτυχθεί στα πλαίσια του έργου και μπορεί να εγκατασταθεί σε ένα κάποιον J2EE application server όπως ο Tomcat, Jetty, JBoss κ.α.. Η επικοινωνία με το API γίνεται με json μηνύματα. Οι χρήστες μπορούν να πειραματιστούν με το API μέσα από την υλοποίηση που έχει γίνει πάνω στο Swagger (https://swagger.io/).

Στη συνέχεια παρουσιάζονται οι βασικές κλήσεις των web services του RESTful API με παραδείγματα μέσω του Swagger UI.

Το API διαθέτει μια σειρά από κλήσεις που φέρνουν τα properties με τα μεταδεδομένα και την πληροφορία για τον χαρακτηρισμό του υλικού. Κάθε χαρακτηριστικό(property) έχει ένα όνομα(mkey) και μια τιμή(value) και ο χρήστης μπορεί να ζητήσει τα ονόματα των properties ξεχωριστά για τα terms, clues και quizzes ή να φέρει τις τιμές για ένα συγκεκριμένο property με βάσει το όνομα.

Στη παρακάτω εικόνα φαίνονται οι κλήσεις για τα properties.

Properties Key-value properties of clues, terms and quizzes

GET	/api/properties/clues/keys	Get all keys that a clue property can have
GET	/api/properties/quizzes/keys	Get all keys that a quiz property can have
GET	/api/properties/terms/keys	Get all keys that a term property can have
GET	/api/properties/values	Get all values of a specific property key
GET	/api/properties/values/clues	Get values of a specific property key for clues that have a set of properties
GET	/api/properties/values/quizzes	Get values of a specific property key for quizzes that have a set of properties
GET	/api/properties/values/terms	Get values of a specific property key for terms that have a set of properties

Στην παρακάτω εικόνα παρουσιάζεται η κλήση του API για την αναζήτηση των τιμών του property με όνομα “ xtizo_lekseis_prefix_word_semantic ” όπου επιστρέφει τις ερμηνείες σύνθετων λέξεων.

Request URL

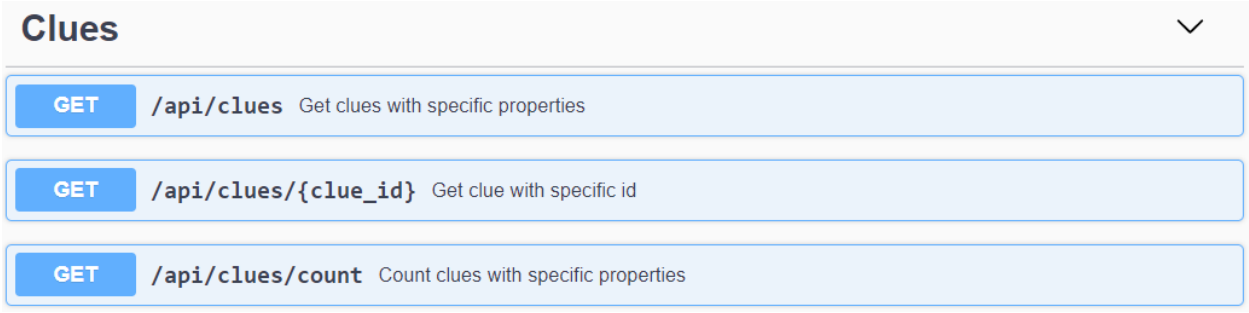
```
http://localhost:8080/lexipaignio-restful-ws/api/properties/values/terms?key=xtizo_lekseis_prefix_word_semantic
```

Server response

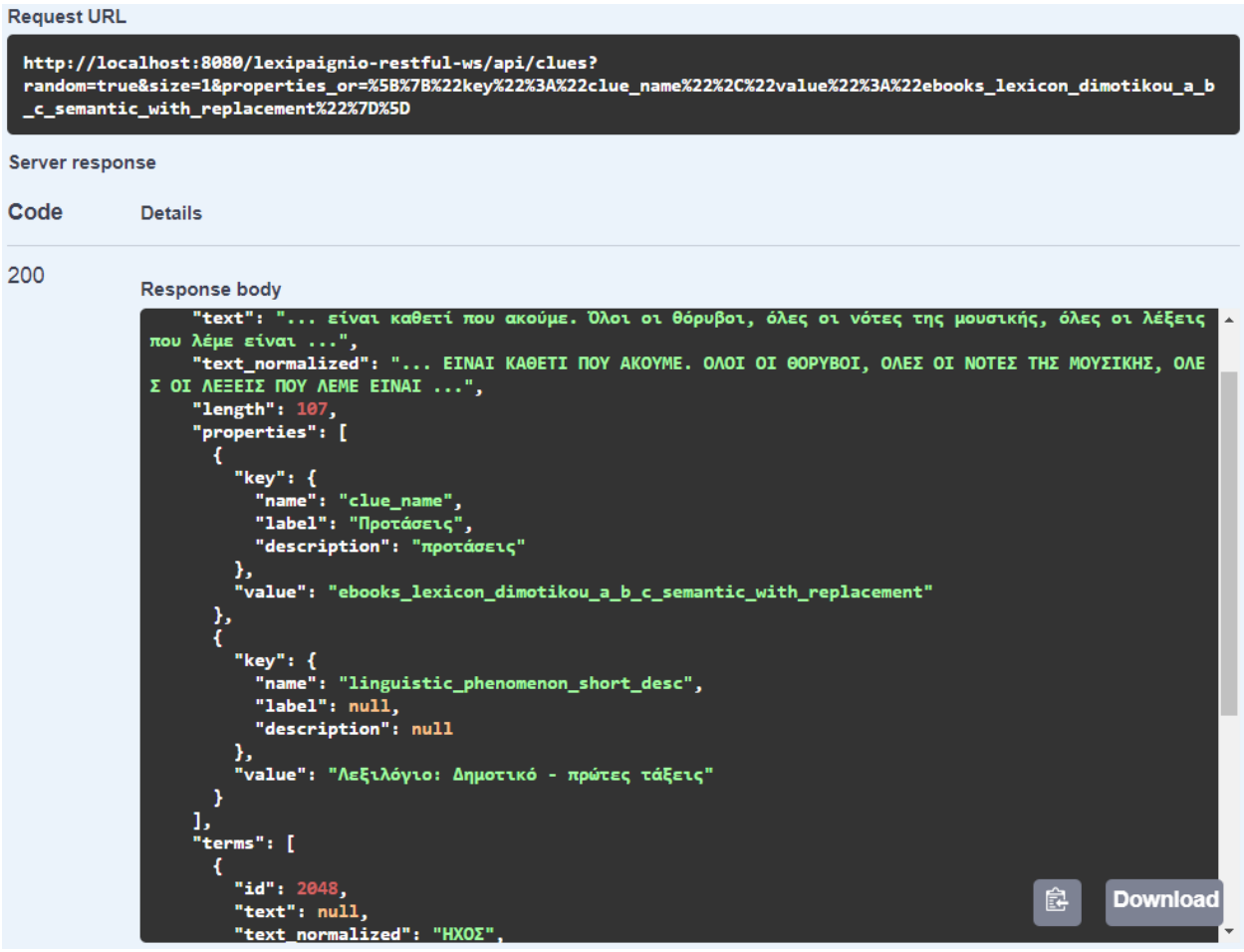
Code	Details
200	Response body


```
[
  "η [{νανοηλεκτρονική}] είναι ο επιστημονικός κλάδος της ηλεκτρονικής που συνδέεται με τη νανοτεχνολογία",
  "το [{νανορομπότ}] είναι μικροσκοπικό ρομπότ που κατασκευάζεται από τις νανοεπιστήμες.",
  "το [{οικόσημο}] είναι το έμβλημα μιας παλαιάς αριστοκρατικής οικογένειας.",
  "η [{επιμόρφωση}] είναι συμπληρωματική κατάρτιση πάνω σε ένα τομέα.",
  "[{κωλοφωτιά}] λέμε επίσης τον πολύ έξυπνο άνθρωπο.",
  "η [{γαστριμαργία}] είναι η αγάπη για το εκλεκτό φαγητό.",
  "κάποιος είναι [{ανίκανος}] όταν δε διαθέτει ικανότητες σε συγκεκριμένο τομέα",
  "η [{αναρχία}] είναι η έλλειψη τάξης.",
  "η [{παραμάνα}] αναλαμβάνει το μεγάλωμα παιδιών που δεν είναι δικά της.",
  "το [{περιεχόμενο}] ενός κουτιού είναι ό,τι υπάρχει μέσα σε αυτό.",
  "για τους ορθόδοξους χριστιανούς το [{Ψυχοσάββατο}] είναι η μέρα όπου μνημονεύουν τους νεκρούς.",
  "η [{παλιοπαρέα}] είναι παρέα αγαπημένων φίλων που γνωρίζονται από παλιά.",
  "ο [{πρωθυπουργός}] είναι ο πρόεδρος της κυβέρνησης και του υπουργικού συμβουλίου",
  "ο [{πρωτομάστορας}] είναι ο επικεφαλής μιας ομάδας μαστόρων. Κάποια από αυτά τα ουσιαστικά αποτελούν τιμητικούς τίτλους (π.χ. {{πρωθυπουργός}}, {{πρωτοπρεσβύτερος}}).",
  "ο [{αυτοδίδακτος}] έχει διδαχθεί κάτι μόνος του χωρίς τη βοήθεια άλλου",
  "ο [{αυτοεξόριστος}] έχει εξοριστεί με τη δική του θέληση σε ξένη χώρα.",
  "ο [{αρχέγονος}] είναι αυτός που υπάρχει από την αρχή της δημιουργίας.",
  "ο [{λιποτάκτης}] είναι αυτός που εγκαταλείπει τις στρατιωτικές τάξεις.",
  "το [{οστεοφυλάκιο}] είναι ειδική θήκη για τη φύλαξη των οστών νεκρού.",
  "ο [{μεταμοντερνισμός}] στη ζωγραφική προτείνει πρωτότυπους συνδυασμούς μοντέρνων και κλασικών στοιχείων με στόχο τον εντυπωσιασμό.",
  "η [{γαστραλγία}] είναι ο πόνος στην περιοχή της κοιλιάς.",
  "η [{οστεοπόρωση}] είναι πάθηση κατά την οποία αραιώνει ο ιστός των οστών, με συνέπ
  αυτ
  Download
νται εύθραυστα",
```

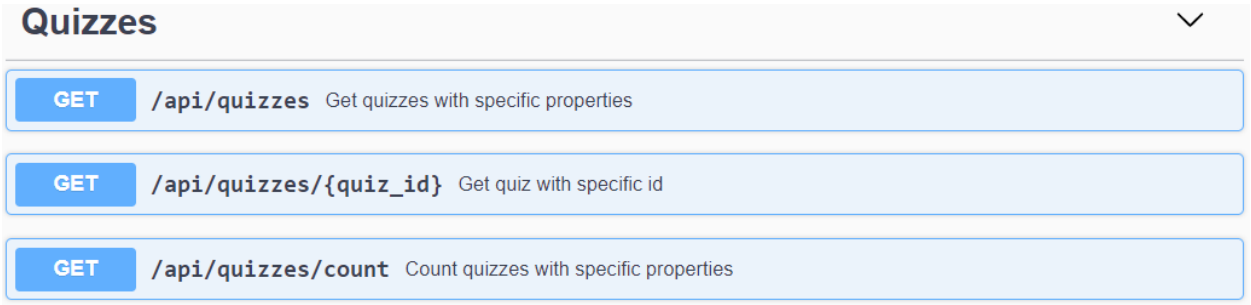
Στη παρακάτω εικόνα φαίνονται οι κλήσεις για τα clues.



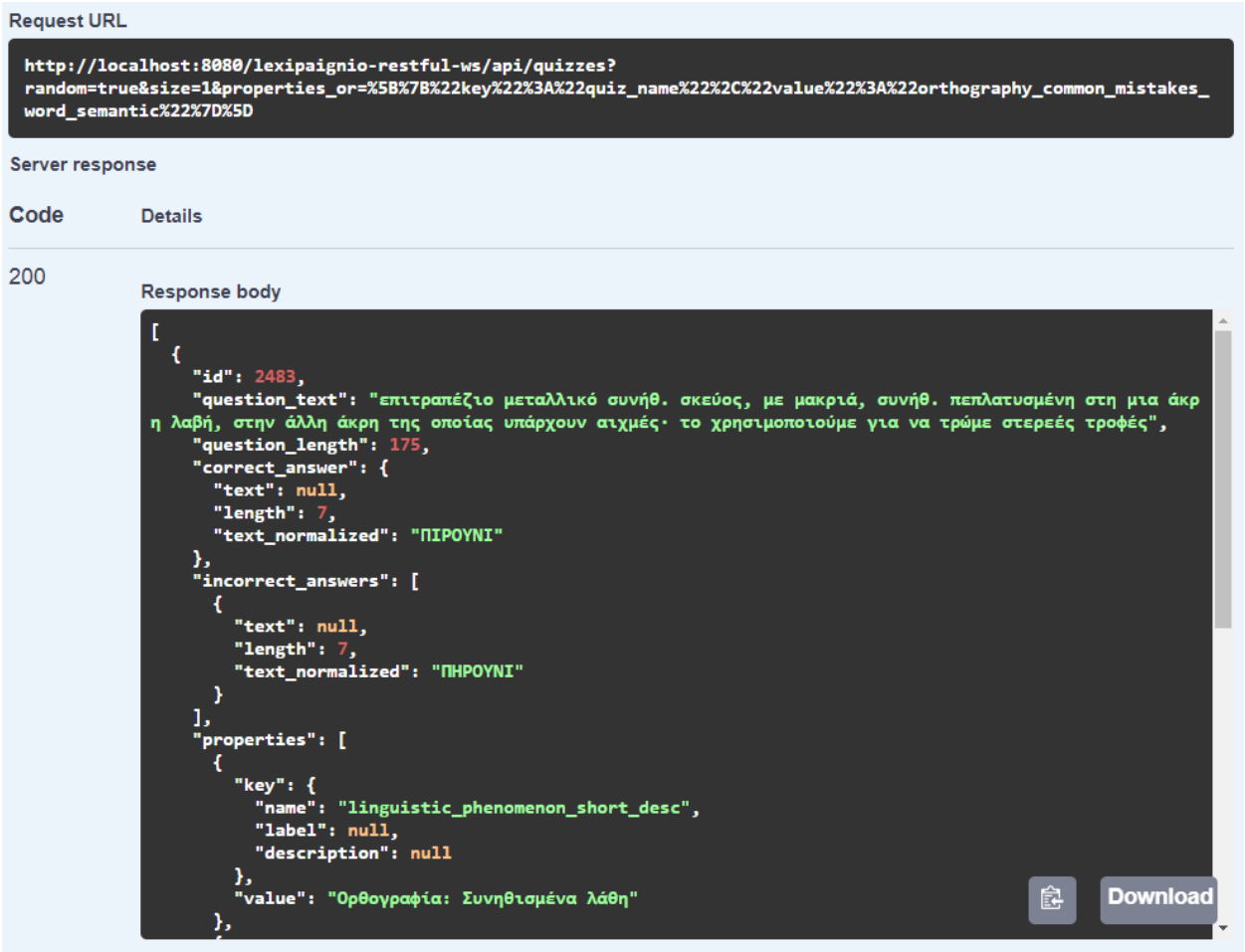
Στην παρακάτω εικόνα παρουσιάζεται η κλήση του API για την αναζήτηση clue που έχει property με όνομα " clue_name" και τιμή "ebooks_lexicon_dimotikou_a_b_c_semantic_with_replacement" όπου επιστρέφει ένα clue με την ερμηνεία της λέξης "ΗΧΟΣ" από το λεξικό της Α-Β-Γ τάξης δημοτικού.



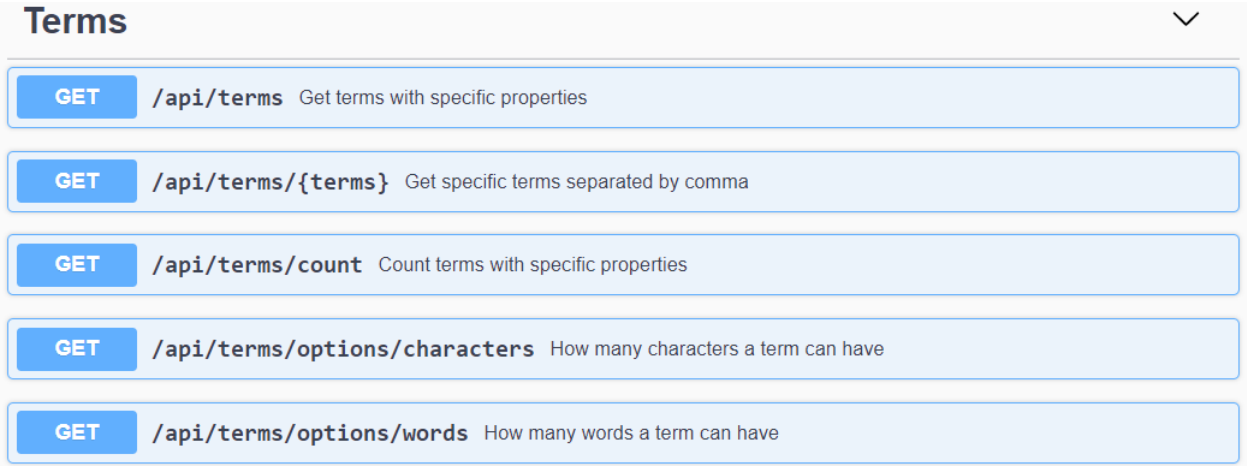
Στη παρακάτω εικόνα φαίνονται οι κλήσεις για τα quizzes.



Στην παρακάτω εικόνα παρουσιάζεται η κλήση του API για την αναζήτηση quiz που έχει property με όνομα “ quiz_name” και τιμή “ orthography_common_mistakes_word_semantic ” όπου επιστρέφει ένα quiz με την ερμηνεία της λέξης “ΠΙΡΟΥΝΙ” ως σωστή απάντηση και με λανθασμένη επιλογή τη λέξη “ΠΗΡΟΥΝΙ”.



Στη παρακάτω εικόνα φαίνονται οι κλήσεις για τα terms.



Στην παρακάτω εικόνα παρουσιάζεται η κλήση του API για την αναζήτηση ενός λεκτικού που έχει property με όνομα " xtizo_lekseis_prefix_comment_desc_word_mention" όπου επιστρέφει τη σύνθετη λέξη "ΟΣΤΕΟΦΥΛΑΚΕΙΟ" με ένα παράδειγμα για την ερμηνεία της από το λεξικό "Χτίζω Λέξεις".

Request URL

```
http://localhost:8080/lexipaignio-restful-ws/api/terms?
random=true&size=1&term_properties_or=%5B%7B%22key%22%3A%22xtizo_lekseis_prefix_comment_desc_word_mention%22%7D%5D
```

Server response

Code	Details
200	Response body


```
    "text_normalized": "ΟΣΤΕΟΦΥΛΑΚΙΟ",
    "length": 12,
    "words_count": 1,
    "properties": [
      {
        "key": {
          "name": "xtizo_lekseis_prefix_semantic_title",
          "label": null,
          "description": null
        },
        "value": "Σχέση με τα οστά"
      },
      {
        "key": {
          "name": "xtizo_lekseis_prefix_comment_desc_word_mention",
          "label": null,
          "description": null
        },
        "value": "Σπανιότερες είναι οι λέξεις με το [[οστεο-]] στο γενικό λεξιλόγιο. Για παράδειγμα,
το [{οστεοφυλάκιο}] είναι ειδική θήκη για τη φύλαξη των οστών νεκρού."
      },
      {
        "key": {
          "name": "xtizo_lekseis_prefix_word_a_component_norma",
          "label": null,
          "description": null
        },
        "value": "ΟΣΤΕΟ"
```


Download

Binary λεξικά, word embeddings

Η δημιουργία του ελληνικού σώματος κειμένων από διαφορετικά ήδη πηγών όπως ειδησιογραφία, ηλεκτρονικά βιβλία, βικιπαίδεια, μέσα κοινωνικής δικτύωσης, υπότιτλοι κινηματογραφικών ταινιών και τηλεοπτικών σειρών, ιστολόγια, κ.α., μας έδωσε την δυνατότητα να δημιουργήσουμε έναν αριθμό από λεκτικές ενσωματώσεις (word embeddings) σε μορφή γλωσσικών πόρων. Η διαδικασία κατασκευής όπως περιγράψαμε είχε δύο φάσεις. Στην 1η φάση κατασκευάζουμε σε μορφή αρχείων κειμένου το λεξιλόγιο με τις συχνότητες εμφάνισης κάθε λέξεις καθώς και τα αρχεία που περιέχουν τα ανύσματα των λέξεων σε ένα πολυδιάστατο χώρο. Στην 2η φάση από το λεξιλόγιο και τα ανύσματα λέξεων κατασκευάζουμε λεξικά σε δυαδική μορφή με κατάλληλα ευρετήρια για αποδοτική αξιοποίηση τους. Για την δημιουργία των αρχείων της 1ης φάσης χρησιμοποιήθηκαν οι εφαρμογές που προσφέρουν οι ερευνητές των αλγορίθμων που είναι γνωστοί με τα ονόματα CBOW, SKIPGRAM και GLOVE. Οι δύο πρώτοι αλγόριθμοι έχουν υλοποιηθεί στην εφαρμογή word2vec και υλοποιούν την δουλειά των Mikolov[17] et. al. Το εργαλείο word2vec[18] έχει ενσωματωθεί και τρέχει μέσα από το περιβάλλον του Mnemosyne. Ο τρίτος αλγόριθμος παρουσιάζεται στη δουλειά των Pennington[19] et. al. και υλοποιείται στη εφαρμογή glove[20]. Οι διαστάσεις όλων των διανυσμάτων που δημιουργήσαμε ήταν 300. Στην περίπτωση του word2vec έχουμε δημιουργήσει και τα ανύσματα των φράσεων. Οι φράσεις δημιουργούνται σε μία ξεχωριστή 3η φάση όπου τα συχνά ζεύγη λέξεων ενώνονται και δημιουργούν μια τεχνητή λέξη για την οποία στην συνέχεια το εργαλείο θα φτιάξει το άνυσμα του.

Συγκεκριμένα, οι λεξικολογικοί πόροι που δημιουργήθηκαν από την παραπάνω διαδικασία είναι:

- vocab_w2v_nr.txt, cbow_300_nr.txt, grcorpus_cbow.dic όπου το vocab_w2v_nr.txt περιέχει το λεξιλόγιο του σώματος κειμένων με τις συχνότητες των λέξεων. Κάθε σειρά του αρχείου έχουμε μία λέξη και την συχνότητα εμφάνισης χωρισμένη με κενό χαρακτήρα. Το w2v είναι ενδεικτικό του αλγόριθμου word2vec και το "nr" δηλώνει ότι οι λέξεις είναι κανονικοποιημένες (normalized) που σημαίνει ότι έχουν όλα τα γράμματα έχουν μετατραπεί σε κεφαλαία άτονα. Το cbow_300_nr.txt δημιουργήθηκε επίσης από το εργαλείο word2vec με εφαρμογή του αλγόριθμου "Continuous Bag Of Words". Το 300 εκφράζει την διάσταση των ανυσμάτων και το. Το αρχείο grcorpus_cbow.dic έχει δημιουργηθεί από το εργαλείο DataCompiler του Mnemosyne και είναι το δυαδικό λεξικό που περιέχει ένα ευρετήριο των λέξεων του λεξιλογίου, την συχνότητα κάθε λέξης και το άνυσμα των 300 αριθμών για την κάθε λέξη.

- vocab_w2v_nr.txt, skipgrm_300_nr.txt, grcorpus_skipgrm.dic είναι τα αρχεία ανυσμάτων για τον αλγόριθμο "Skip-Gram" του εργαλείου word2vec. Το λεξιλόγιο και οι συχνότητες των λέξεων είναι το ίδιο με αυτό του CBOW που αναφέραμε προηγουμένως. Το αρχείο grcorpus_skipgrm.dic είναι το δυαδικό λεξικό που ενσωματώνει το άνυσμα μαζί με το ευρετήριο όπου είδαμε στη προηγούμενη περίπτωση.

- vocab_glove_nr.txt, glove_300_nr.txt, grcorpus_glove.dic είναι τα τρία αντίστοιχα αρχεία για τον αλγόριθμο GLOVE, δηλ. το λεξιλόγιο με τις συχνότητες, το αρχείο με τις ενσωματώσεις λέξεων όπως δημιουργήθηκε από το εργαλείο glove και το δυαδικό λεξικό που δημιουργήθηκε από το Mnemosyne (DataCompiler) και μπορεί να χρησιμοποιηθεί από αναλυτές του Mnemosyne.

[17] https://arxiv.org/abs/1301.3781
[18] https://github.com/tmikolov/word2vec
[19] https://nlp.stanford.edu/pubs/glove.pdf
[20] https://github.com/stanfordnlp/GloVe

- vocab_grphrase_nr.txt, grphrase_cbow_300_nr.txt, grphrase_cbow.dic είναι τα αρχεία που δημιουργήθηκαν με την επεξεργασία φράσεων. Τα δύο πρώτα αρχεία, δηλ. το vocab_grphrase_nr.txt και το grphrase_cbow_300_nr.txt δημιουργήθηκαν από το εργαλείο word2vec με την επεξεργασία του σώματος για την εύρεση συχνών φράσεων και την διαχείρισή τους ως λέξεις. Στο λεξιλόγιο θα δούμε εγγραφές όπως οι «ΜΕΡΙΚΕΣ_ΦΟΡΕΣ 107070», «ΕΥΡΩΠΑΪΚΟΥ_ΚΟΙΝΟΒΟΥΛΙΟΥ 98337». Ο αλγόριθμος που χρησιμοποιήθηκε είναι ο CBOW και το δυαδικό λεξικό που δημιούργησε το Mnemosyne έχει ως λέξεις τις φράσεις (ενωμένες με '_' λέξεις) του λεξιλογίου.
- vocab_grphrase_nr.txt, grphrase_skipgrm_300_nr.txt, grphrase_skipgrm.dic είναι τα αρχεία για τις φράσεις με τον αλγόριθμο του SKIPGRAM. Όπως και στην περίπτωση των απλών λέξεων τα λεξιλόγια των CBOW και SKIPGRAM για τις φράσεις είναι τα ίδια μιας και παρήχθησαν από το word2vec πριν την δημιουργία των ανυσμάτων.

5. Παραρτήματα

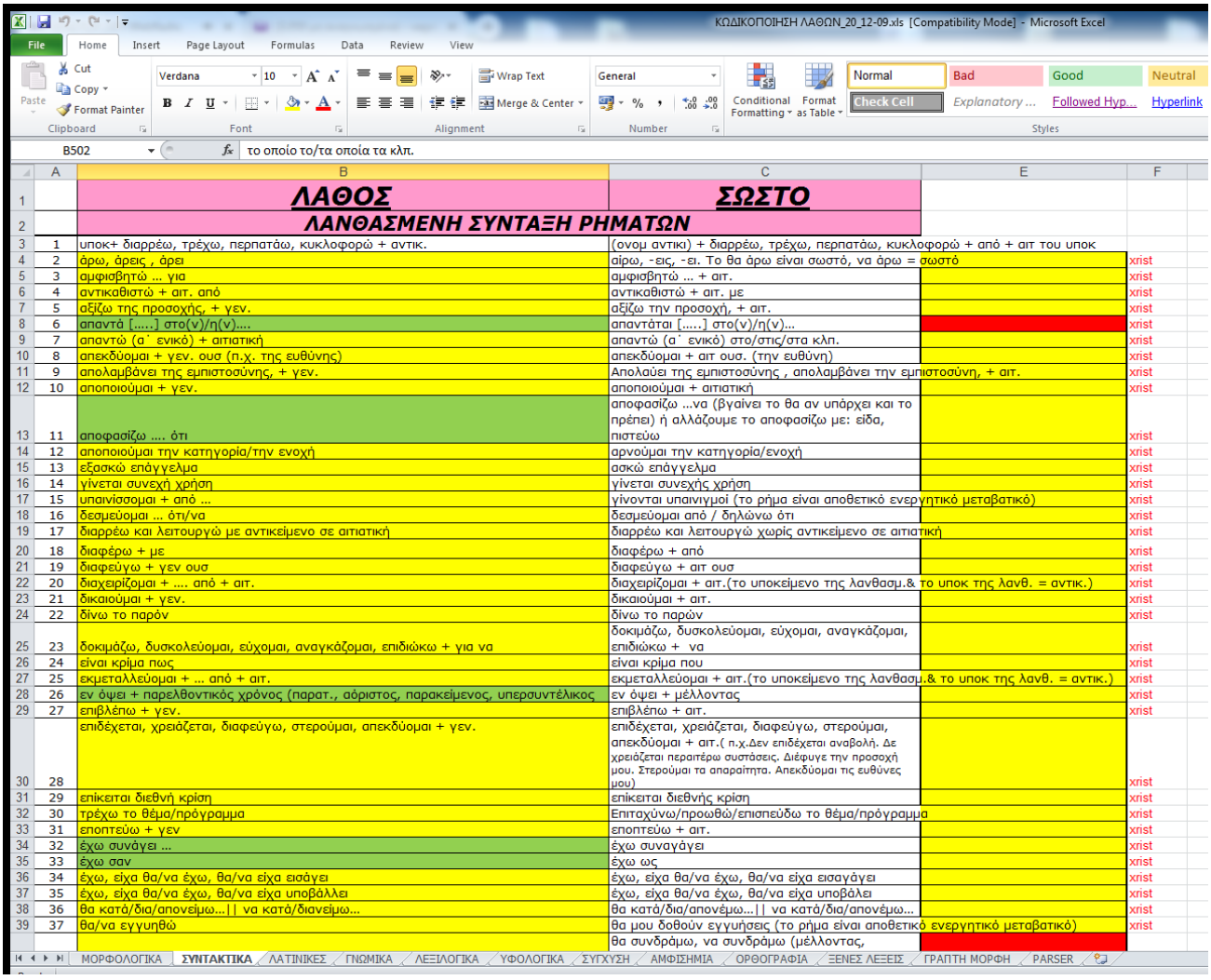
1.1. Γραμματικός Διορθωτής

Κανόνες περιγραφής λαθών

Στο πλαίσιο του γραμματικού διορθωτή, έχουν καταγραφεί λάθη διαφόρων κατηγοριών (Μορφολογικά, Συντακτικά, Λατινικά, Γνωμικά, κοκ), τα οποία χρησιμοποιούνται για τη λειτουργία του (βλέπε σχετικό παράρτημα στο τέλος του παραδοτέου).

Οι κατηγορίες και οι δομές αυτές, αναλύθηκαν και αξιοποιήθηκαν και για τα παιχνίδια του «Λεξιπαιγνίου».

Παρακάτω φαίνεται μια εικόνα των κανόνων, αυτών, το σύνολο των οποίων περιλαμβάνονται στο παραδοτέο Π3.3.

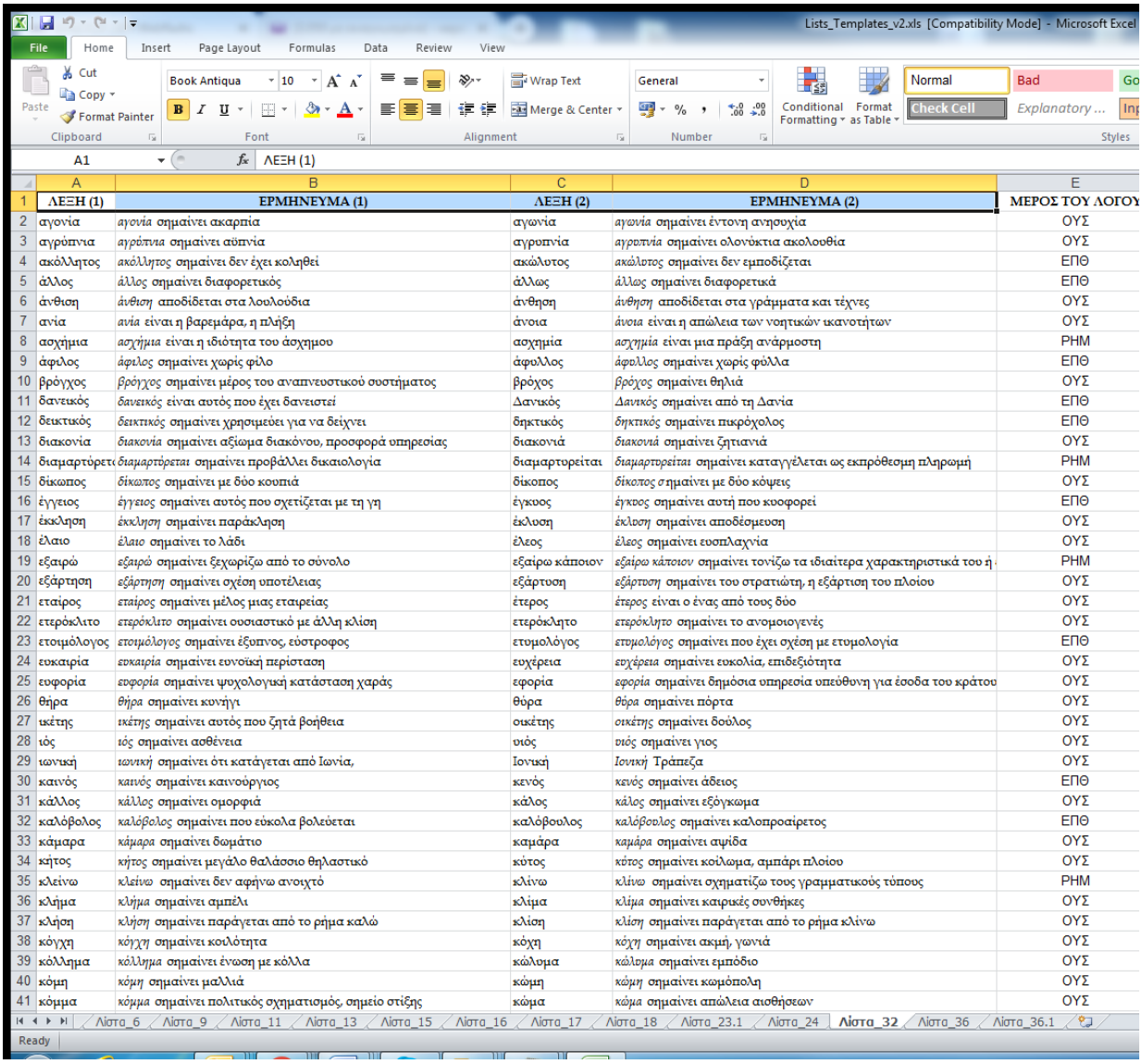


ΚΩΔΙΚΟΠΟΙΗΣΗ ΛΑΘΩΝ_20_12-09.xls [Compatibility Mode] - Microsoft Excel

		ΛΑΘΟΣ	ΣΩΣΤΟ		
		ΛΑΝΘΑΣΜΕΝΗ ΣΥΝΤΑΞΗ ΡΗΜΑΤΩΝ			
3	1	υποκ+ διαρρέω, τρέχω, περπατάω, κυκλοφορώ + αντικ.	(ονομ αντικ) + διαρρέω, τρέχω, περπατάω, κυκλοφορώ + από + αιτ του υποκ		
4	2	άρω, άρεις , άρει	αίρω, -εις, -ει. Το θα άρω είναι σωστό, να άρω = σωστό		xrist
5	3	αμφισβητώ ... για	αμφισβητώ ... + αιτ.		xrist
6	4	αντικαθιστώ + αιτ. από	αντικαθιστώ + αιτ. με		xrist
7	5	αξίζω της προσοχής, + γεν.	αξίζω την προσοχή, + αιτ.		xrist
8	6	απαντά [.....] στο(ν)/η(ν)....	απαντάται [.....] στο(ν)/η(ν)...		xrist
9	7	απαντώ (α΄ ενικό) + αιτιατική	απαντώ (α΄ ενικό) στο/στις/στα κλπ.		xrist
10	8	απεκδύομαι + γεν. ουσ (π.χ. της ευθύνης)	απεκδύομαι + αιτ ουσ. (την ευθύνη)		xrist
11	9	απολαμβάνει της εμπιστοσύνης, + γεν.	Απολαύει της εμπιστοσύνης , απολαμβάνει την εμπιστοσύνη, + αιτ.		xrist
12	10	αποποιούμαι + γεν.	αποποιούμαι + αιτιατική		xrist
13	11	αποφασίζω .... ότι	αποφασίζω ...να (βγαίνει το θα αν υπάρχει και το πρέπει) ή αλλάζουμε το αποφασίζω με: είδα, πιστεύω		xrist
14	12	αποποιούμαι την κατηγορία/την ενοχή	αρνούμαι την κατηγορία/ενοχή		xrist
15	13	εξασκώ επάγγελμα	ασκώ επάγγελμα		xrist
16	14	γίνεται συνεχή χρήση	γίνεται συνεχής χρήση		xrist
17	15	υπαινίσσομαι + από ...	γίνονται υπαινιγμοί (το ρήμα είναι αποθετικό ενεργητικό μεταβατικό)		xrist
18	16	δεσμεύομαι ... ότι/να	δεσμεύομαι από / δηλώνω ότι		xrist
19	17	διαρρέω και λειτουργώ με αντικείμενο σε αιτιατική	διαρρέω και λειτουργώ χωρίς αντικείμενο σε αιτιατική		xrist
20	18	διαφέρω + με	διαφέρω + από		xrist
21	19	διαφεύγω + γεν ουσ	διαφεύγω + αιτ ουσ		xrist
22	20	διαχειρίζομαι + ... από + αιτ.	διαχειρίζομαι + αιτ.(το υποκείμενο της λανθασμ.& το υποκ της λανθ. = αντικ.)		xrist
23	21	δικαιούμαι + γεν.	δικαιούμαι + αιτ.		xrist
24	22	δίνω το παρόν	δίνω το παρών		xrist
25	23	δοκιμάζω, δυσκολεύομαι, εύχομαι, αναγκάζομαι, επιδιώκω + για να	δοκιμάζω, δυσκολεύομαι, εύχομαι, αναγκάζομαι, επιδιώκω + να		xrist
26	24	είναι κρίμα πως	είναι κρίμα που		xrist
27	25	εκμεταλλεύομαι + ... από + αιτ.	εκμεταλλεύομαι + αιτ.(το υποκείμενο της λανθασμ.& το υποκ της λανθ. = αντικ.)		xrist
28	26	εν όψει + παρελθοντικός χρόνος (παρατ., αόριστος, παρακείμενος, υπερσυντέλικος	εν όψει + μέλλοντας		xrist
29	27	επιβλέπω + γεν.	επιβλέπω + αιτ.		xrist
30	28	επιδέχεται, χρειάζεται, διαφεύγω, στερούμαι, απεκδύομαι + γεν.	επιδέχεται, χρειάζεται, διαφεύγω, στερούμαι, απεκδύομαι + αιτ.(π.χ.Δεν επιδέχεται αναβολή. Δε χρειάζεται περαιτέρω συστάσεις. Διέφυγε την προσοχή μου. Στερούμαι τα απαραίτητα. Απεκδύομαι τις ευθύνες μου)		xrist
31	29	επίκειται διεθνή κρίση	επίκειται διεθνής κρίση		xrist
32	30	τρέχω το θέμα/πρόγραμμα	Επιταχύνω/προωθώ/επισπεύδω το θέμα/πρόγραμμα		xrist
33	31	εποπτεύω + γεν	εποπτεύω + αιτ.		xrist
34	32	έχω συνάγει ...	έχω συναγάγει		xrist
35	33	έχω σαν	έχω ως		xrist
36	34	έχω, είχα θα/να έχω, θα/να είχα εισάγει	έχω, είχα θα/να έχω, θα/να είχα εισαγάγει		xrist
37	35	έχω, είχα θα/να έχω, θα/να είχα υποβάλλει	έχω, είχα θα/να έχω, θα/να είχα υποβάλει		xrist
38	36	θα κατά/δια/απονείμω...|| να κατά/διανείμω...	θα κατά/δια/απονέμω...|| να κατά/δια/απονέμω...		xrist
39	37	θα/να εγγυηθώ	θα μου δοθούν εγγυήσεις (το ρήμα είναι αποθετικό ενεργητικό μεταβατικό)		xrist
			θα συνδράμω, να συνδράμω (μέλλοντας,		

ΜΟΡΦΟΛΟΓΙΚΑ | ΣΥΝΤΑΚΤΙΚΑ | ΛΑΤΙΝΙΚΕΣ | ΓΝΩΜΙΚΑ | ΛΕΞΙΛΟΓΙΚΑ | ΥΦΟΛΟΓΙΚΑ | ΣΥΓΧΥΣΗ | ΑΜΦΙΣΗΜΙΑ | ΟΡΘΟΓΡΑΦΙΑ | ΞΕΝΕΣ ΛΕΞΕΙΣ | ΓΡΑΠΤΗ ΜΟΡΦΗ | PARSER

Επιπλέον, στο πλαίσιο του γραμματικού διορθωτή, έχουν καταγραφεί μια σειρά από λάθη (με την αντίστοιχη ορθή «διόρθωση») – βλέπε επόμενη εικόνα, που και αυτά αξιοποιήθηκαν στο πλαίσιο του Λεξιπαιγνίου.



Lists_Templates_v2.xls [Compatibility Mode] - Microsoft Excel

	ΛΕΞΗ (1)	ΕΡΜΗΝΕΥΜΑ (1)	ΛΕΞΗ (2)	ΕΡΜΗΝΕΥΜΑ (2)	ΜΕΡΟΣ ΤΟΥ ΛΟΓΟΥ
2	αγονία	*αγονία* σημαίνει ακαρπία	αγωνία	*αγωνία* σημαίνει έντονη ανησυχία	ΟΥΣ
3	αγρόπνια	*αγρόπνια* σημαίνει αϋπνία	αγροπνία	*αγροπνία* σημαίνει ολονύκτια ακολουθία	ΟΥΣ
4	ακόλλητος	*ακόλλητος* σημαίνει δεν έχει κοληθεί	ακώλυτος	*ακώλυτος* σημαίνει δεν εμποδίζεται	ΕΠΘ
5	άλλος	*άλλος* σημαίνει διαφορετικός	άλλως	*άλλως* σημαίνει διαφορετικά	ΕΠΘ
6	άνθιση	*άνθιση* αποδίδεται στα λουλούδια	άνθηση	*άνθηση* αποδίδεται στα γράμματα και τέχνες	ΟΥΣ
7	ανία	*ανία* είναι η βαρεμάρα, η πλήξη	άνοια	*άνοια* είναι η απώλεια των νοητικών ικανοτήτων	ΟΥΣ
8	ασχήμια	*ασχήμια* είναι η ιδιότητα του άσχημου	ασχημία	*ασχημία* είναι μια πράξη ανάρμοστη	ΡΗΜ
9	άφιλος	*άφιλος* σημαίνει χωρίς φίλο	άφυλλος	*άφυλλος* σημαίνει χωρίς φύλλα	ΕΠΘ
10	βρόγχος	*βρόγχος* σημαίνει μέρος του αναπνευστικού συστήματος	βρόχος	*βρόχος* σημαίνει θηλιά	ΟΥΣ
11	δανεικός	*δανεικός* είναι αυτός που έχει δανειστεί	Δανικός	*Δανικός* σημαίνει από τη Δανία	ΕΠΘ
12	δεικτικός	*δεικτικός* σημαίνει χρησιμεύει για να δείχνει	δηκτικός	*δηκτικός* σημαίνει πικρόχολος	ΕΠΘ
13	διακονία	*διακονία* σημαίνει αξίωμα διακόνου, προσφορά υπηρεσίας	διακονιά	*διακονιά* σημαίνει ζητιανιά	ΟΥΣ
14	διαμαρτύρετι	*διαμαρτύρεται* σημαίνει προβάλλει δικαιολογία	διαμαρτορείται	*διαμαρτυρείται* σημαίνει καταγγέλεται ως εκπρόθεσμη πληρωμή	ΡΗΜ
15	δίκωπος	*δίκωπος* σημαίνει με δύο κουπιά	δίκοπος	*δίκοπος σ*ημαίνει με δύο κόψεις	ΟΥΣ
16	έγγειος	*έγγειος* σημαίνει αυτός που σχετίζεται με τη γη	έγκυος	*έγκυος* σημαίνει αυτή που κυοφορεί	ΕΠΘ
17	έκκληση	*έκκληση* σημαίνει παράκληση	έκλυση	*έκλυση* σημαίνει αποδέσμευση	ΟΥΣ
18	έλαιο	*έλαιο* σημαίνει το λάδι	έλεος	*έλεος* σημαίνει ευσπλαχνία	ΟΥΣ
19	εξαιρώ	*εξαιρώ* σημαίνει ξεχωρίζω από το σύνολο	εξαίρω κάποιον	*εξαίρω κάποιον* σημαίνει τονίζω τα ιδιαίτερα χαρακτηριστικά του ή	ΡΗΜ
20	εξάρτηση	*εξάρτηση* σημαίνει σχέση υποτέλειας	εξάρτυση	*εξάρτυση* σημαίνει του στρατιώτη, η εξάρτιση του πλοίου	ΟΥΣ
21	εταίρος	*εταίρος* σημαίνει μέλος μιας εταιρείας	έτερος	*έτερος* είναι ο ένας από τους δύο	ΟΥΣ
22	ετερόκλιτο	*ετερόκλιτο* σημαίνει ουσιαστικό με άλλη κλίση	ετερόκλητο	*ετερόκλητο* σημαίνει το ανομοιογενές	ΟΥΣ
23	ετοιμόλογος	*ετοιμόλογος* σημαίνει έξυπνος, εύστροφος	ετυμολόγος	*ετυμολόγος* σημαίνει που έχει σχέση με ετυμολογία	ΕΠΘ
24	ευκαιρία	*ευκαιρία* σημαίνει ευνοϊκή περίσταση	ευχέρεια	*ευχέρεια* σημαίνει ευκολία, επιδεξιότητα	ΟΥΣ
25	ευφορία	*ευφορία* σημαίνει ψυχολογική κατάσταση χαράς	εφορία	*εφορία* σημαίνει δημόσια υπηρεσία υπεύθυνη για έσοδα του κράτου	ΟΥΣ
26	θήρα	*θήρα* σημαίνει κυνήγι	θύρα	*θύρα* σημαίνει πόρτα	ΟΥΣ
27	ικέτης	*ικέτης* σημαίνει αυτός που ζητά βοήθεια	οικέτης	*οικέτης* σημαίνει δούλος	ΟΥΣ
28	ιός	*ιός* σημαίνει ασθένεια	υιός	*υιός* σημαίνει γιος	ΟΥΣ
29	ιωνική	*ιωνική* σημαίνει ότι κατάγεται από Ιωνία,	Ιονική	*Ιονική* Τράπεζα	ΟΥΣ
30	καινός	*καινός* σημαίνει καινούργιος	κενός	*κενός* σημαίνει άδειος	ΕΠΘ
31	κάλλος	*κάλλος* σημαίνει ομορφιά	κάλος	*κάλος* σημαίνει εξόγκωμα	ΟΥΣ
32	καλόβολος	*καλόβολος* σημαίνει που εύκολα βολεύεται	καλόβουλος	*καλόβουλος* σημαίνει καλοπροαίρετος	ΕΠΘ
33	κάμαρα	*κάμαρα* σημαίνει δωμάτιο	καμάρα	*καμάρα* σημαίνει αψίδα	ΟΥΣ
34	κήτος	*κήτος* σημαίνει μεγάλο θαλάσσιο θηλαστικό	κύτος	*κύτος* σημαίνει κοίλωμα, αμπάρι πλοίου	ΟΥΣ
35	κλείνω	*κλείνω* σημαίνει δεν αφήνω ανοιχτό	κλίνω	*κλίνω* σημαίνει σχηματίζω τους γραμματικούς τύπους	ΡΗΜ
36	κλήμα	*κλήμα* σημαίνει αμπέλι	κλίμα	*κλίμα* σημαίνει καιρικές συνθήκες	ΟΥΣ
37	κλήση	*κλήση* σημαίνει παράγεται από το ρήμα καλώ	κλίση	*κλίση* σημαίνει παράγεται από το ρήμα κλίνω	ΟΥΣ
38	κόγχη	*κόγχη* σημαίνει κοιλότητα	κόχη	*κόχη* σημαίνει ακμή, γωνιά	ΟΥΣ
39	κόλλημα	*κόλλημα* σημαίνει ένωση με κόλλα	κώλυμα	*κώλυμα* σημαίνει εμπόδιο	ΟΥΣ
40	κόμη	*κόμη* σημαίνει μαλλιά	κώμη	*κώμη* σημαίνει κωμόπολη	ΟΥΣ
41	κόμμα	*κόμμα* σημαίνει πολιτικός σχηματισμός, σημείο στίξης	κώμα	*κώμα* σημαίνει απώλεια αισθήσεων	ΟΥΣ

Λίστα_6 | Λίστα_9 | Λίστα_11 | Λίστα_13 | Λίστα_15 | Λίστα_16 | Λίστα_17 | Λίστα_18 | Λίστα_23.1 | Λίστα_24 | **Λίστα_32** | Λίστα_36 | Λίστα_36.1

Ready

Κανόνες διαμόρφωσης γραμματικού διορθωτή

Για τη διαμόρφωση του γραμματικού διορθωτή, με τρόπο ώστε να εντοπίζει τα «τριγενή και δικατάληκτα επίθετα», περιγράφηκαν μια σειρά από κανόνες, που καταγράφονται στη συνέχεια:

section

/* GGC_lexipaignio_adjhs_1*/ // οκ 17/7/2020
{1}

```
[ARULE="lexipaignio_adjhs_1",
EVTEXT=TagEvent("gevent.wrong","GEVENT","error_PAST_TENSE","%f","ERMSG","Ο σχηματισμός της κατάληξης του επιθέτου είναι λανθασμένος. Αντικαταστήστε με την κατάληξη '-η'.")]  =>
                    (
                     [LEXY->HasMAttrs([ART, ACC, SING])]  |
                            [LEXY->HasMAttrs([PREP])]          |
        [LEXY->CanMatch("στον",[ACC, SING])]          // χωριστός κανόνας για τα θηλυκά
                    ),
                            []{0,2}
          \
                            [LEXY->SuffixCanMatch("ης",[ADJ]), TTEXT ->Suffix("ης")]  |
                            [LEXY->SuffixCanMatch("ης",[ADJ]), TTEXT ->Suffix("ες")]  |
                            [LEXY->SuffixCanMatch("ής",[ADJ]), TTEXT ->Suffix("ής")]  |
                            [LEXY->SuffixCanMatch("ής",[ADJ]), TTEXT ->Suffix("ές")]
          /
                            [LEXY->HasMAttrs([ACC, SING, N, NOM, VOC])]
          ;
/* GGC_lexipaignio_adjhs_1_1*/  // οκ 17/7/2020
{1}
[ARULE="lexipaignio_adjhs_1_1",
EVTEXT=TagEvent("gevent.wrong","GEVENT","error_PAST_TENSE","%f","ERMSG","Ο σχηματισμός της κατάληξης του επιθέτου είναι λανθασμένος. Αντικαταστήστε με την κατάληξη '-η'.")]  =>
                            [LEXY->HasMAttrs([PREP])],
                            [LEXY->HasMAttrs([ACC, SING, ADJ, NOM, VOC])],
                            [LEXY->HasMAttrs([ACC, SING, N, NOM, VOC])],
                            []{0,6}
          \
                            [LEXY->SuffixCanMatch("ης",[ADJ]), TTEXT ->Suffix("ης")]  |
                            [LEXY->SuffixCanMatch("ης",[ADJ]), TTEXT ->Suffix("ες")]  |
                            [LEXY->SuffixCanMatch("ής",[ADJ]), TTEXT ->Suffix("ής")]  |
                            [LEXY->SuffixCanMatch("ής",[ADJ]), TTEXT ->Suffix("ές")]
          /
                            [LEXY->HasMAttrs([ACC, SING, N, NOM, VOC])]
          ;
/* GGC_lexipaignio_adjhs_1_3*/  // οκ 17/7/2020 , χωριστός κανόνας για τα αρσενικά ΑΠΑΡΑΙΤΗΤΟ
{1}
[ARULE="lexipaignio_adjhs_1_3",
EVTEXT=TagEvent("gevent.wrong","GEVENT","error_PAST_TENSE","%f","ERMSG","Ο σχηματισμός της κατάληξης του επιθέτου είναι λανθασμένος. Αντικαταστήστε με την κατάληξη '-η'.")]  =>
                            [LEXY->CanMatch("στον",[ACC, SING])],
                            []{0,2}
          \
                            [LEXY->SuffixCanMatch("ης",[ADJ]), TTEXT ->Suffix("ης")]  |
                            [LEXY->SuffixCanMatch("ης",[ADJ]), TTEXT ->Suffix("ες")]  |
                            [LEXY->SuffixCanMatch("ής",[ADJ]), TTEXT ->Suffix("ής")]  |
                            [LEXY->SuffixCanMatch("ής",[ADJ]), TTEXT ->Suffix("ές")]
          /
                            [LEXY->HasMAttrs([ACC, SING, N])]
```

```
        ;
/* GGC_lexipaignio_adjhs_2*/  // οκ 17/7/2020
{1}
[ARULE="lexipaignio_adjhs_2",
EVTEXT=TagEvent("gevent.wrong","GEVENT","error_PAST_TENSE","%f","ERMSG","Ο σχηματισμός της
κατάληξης του επιθέτου είναι λανθασμένος. Αντικαταστήστε με την κατάληξη '-ες'.")]  =>
                (       [LEXY->HasMAttrs([ART, ACC, SING])]  |
                        [LEXY->HasMAttrs([PREP])]
                 )
        \
                        [LEXY->SuffixCanMatch("ης",[ADJ]), TTEXT ->Suffix("η")]   |
                        [LEXY->SuffixCanMatch("ης",[ADJ]), TTEXT ->Suffix("ης")]  |
                        [LEXY->SuffixCanMatch("ής",[ADJ]), TTEXT ->Suffix("ή")]   |
                        [LEXY->SuffixCanMatch("ής",[ADJ]), TTEXT ->Suffix("ής")]

        /
                        [LEXY->HasMAttrs([ACC, NEUT, NOM, VOC, SING])]
        ;
/* GGC_lexipaignio_pollys_1*/  // οκ 19-7-2020
{1}
[ARULE="GGC_lexipaignio_pollys_1", EVTEXT=TagEvent("gevent.wrong","GEVENT",
"AGRREMENT_PART_ADJECTIVE","%f","ERMSG","Στη συγκεκριμένη ονοματική φράση δεν υπάρχει
συμφωνία. Αντικαταστήστε το 'πολλή' με το 'πολύ'.")]  =>
    \
                        [TTEXT->Match("πολλή")]
                /
                        [TTEXT->Match("πιο")]?,
                         (
                                [LEXY->HasMAttrs([MASC, SING, NOM, ADJ])]   |
                           [LEXY->HasMAttrs([MASC, SING, NOM, PART])]
                    )
                ;
/* GGC_lexipaignio_pollys_2*/  // οκ 19-7-2020
{1}
[ARULE="GGC_lexipaignio_pollys_2", EVTEXT=TagEvent("gevent.wrong","GEVENT",
"AGRREMENT_PART_ADJECTIVE","%f","ERMSG","Αντικαταστήστε το 'πολλή' με το 'πολύ'.")]  =>
       [LEXY->HasMAttrs([V])],
                  []{0,2}
                \
                        [TTEXT->Match("πολλή")]
                /
                        [LEXY->HasMAttrs([ACC, ART])],
        (
                          [LEXY->HasMAttrs([ACC, N])]     |
         [LEXY->HasMAttrs([ACC, ADJ])]   |
                        [LEXY->HasMAttrs([ACC, PART])]
        )
                ;
```

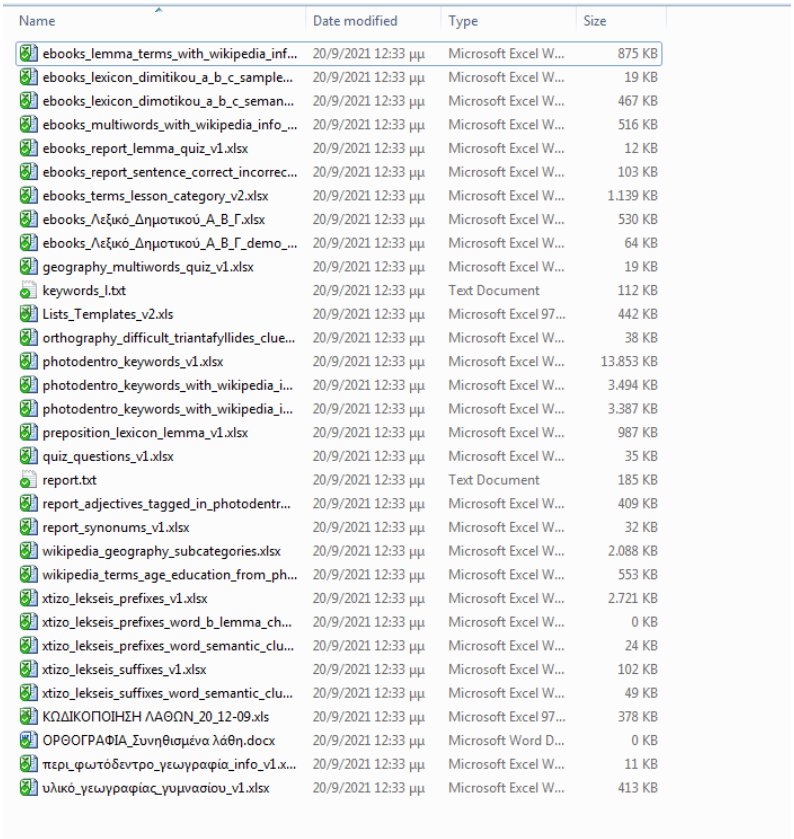
```
/* GGC_lexipaignio_pollys_2_1*/  // οκ 19-7-2020
{2}
[ARULE="GGC_lexipaignio_pollys_2_1", EVTEXT=TagEvent("gevent.wrong","GEVENT",
"AGRREMENT_PART_ADJECTIVE","%f","ERMSG","Αντικαταστήστε το 'πολλή' με το 'πολύ'.")]  =>
        [LEXY->HasMAttrs([V])],
                  []{0,2}
                \
                        [TTEXT->Match("πολλή")]
                /
                        [LEXY->HasMAttrs([ACC, ART, FEM]), ONTO?=$x:GNC_Agreement(1,[ART])],
                   [LEXY->HasMAttrs([ACC, N, FEM]), ONTO?=$x:GNC_Agreement(1,[N])]
                ;

/* GGC_lexipaignio_pollys_2_1*/  // οκ 19-7-2020 // μεταφέρουν πολύ νερό (γι αυτό έσβησα το άρθρο,
σε σχέση με τον προηγούμενο κανόνα)
{1}
[ARULE="GGC_lexipaignio_pollys_2_1", EVTEXT=TagEvent("gevent.wrong","GEVENT",
"AGRREMENT_PART_ADJECTIVE","%f","ERMSG","Αντικαταστήστε το 'πολλή' με το 'πολύ'.")]  =>
        [LEXY->HasMAttrs([V, PLUR, C_P])],
        []{0,2}
                \
                        [TTEXT->Match("πολλή")]
                /
                        [LEXY->HasMAttrs([ACC, N])]
                ;

end
```

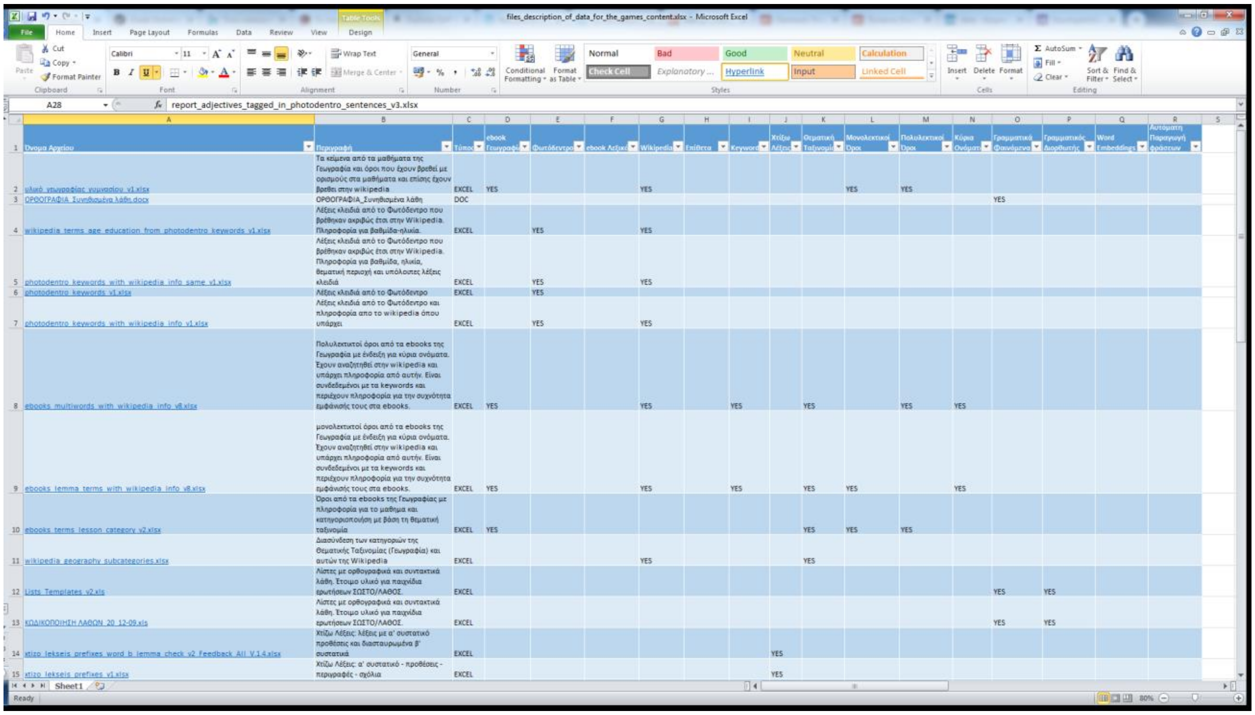
1.2. Αποτελέσματα Επεξεργασίας Δεδομένων

Στο παράρτημα αυτό, καταγράφονται τα αρχεία που προέκυψαν από την επεξεργασία των δεδομένων με σκοπό να αποτελέσουν την είσοδο για τα παιχνίδια που υλοποιούνται. Τα ίδια τα αρχεία περιλαμβάνονται στο Παραδοτέο Π3.3.



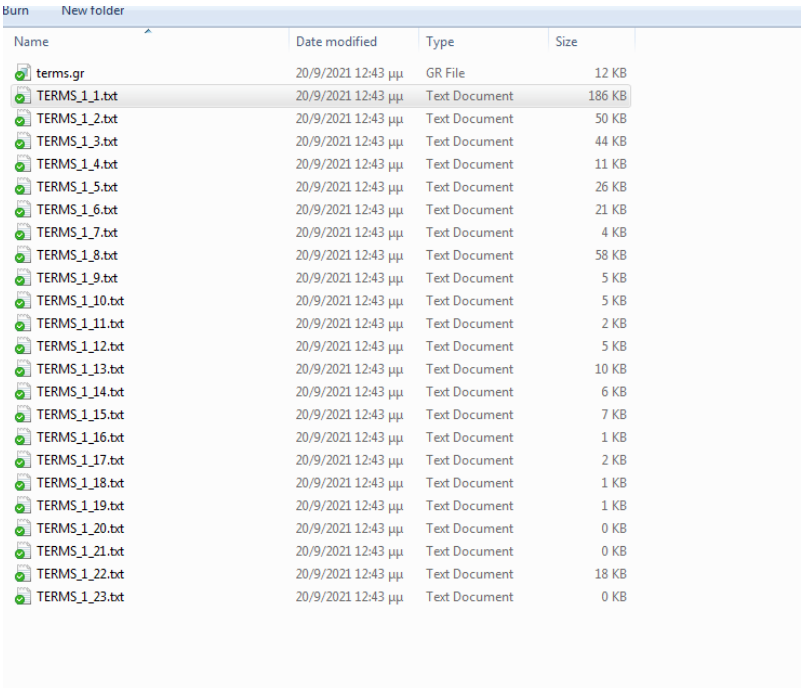
Name	Date modified	Type	Size
ebooks_lemma_terms_with_wikipedia_inf...	20/9/2021 12:33 μμ	Microsoft Excel W...	875 KB
ebooks_lexicon_dimitikou_a_b_c_sample...	20/9/2021 12:33 μμ	Microsoft Excel W...	19 KB
ebooks_lexicon_dimotikou_a_b_c_seman...	20/9/2021 12:33 μμ	Microsoft Excel W...	467 KB
ebooks_multiwords_with_wikipedia_info_...	20/9/2021 12:33 μμ	Microsoft Excel W...	516 KB
ebooks_report_lemma_quiz_v1.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	12 KB
ebooks_report_sentence_correct_incorrec...	20/9/2021 12:33 μμ	Microsoft Excel W...	103 KB
ebooks_terms_lesson_category_v2.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	1.139 KB
ebooks_Λεξικό_Δημοτικού_Α_Β_Γ.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	530 KB
ebooks_Λεξικό_Δημοτικού_Α_Β_Γ_demo_...	20/9/2021 12:33 μμ	Microsoft Excel W...	64 KB
geography_multiwords_quiz_v1.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	19 KB
keywords_l.txt	20/9/2021 12:33 μμ	Text Document	112 KB
Lists_Templates_v2.xls	20/9/2021 12:33 μμ	Microsoft Excel 97...	442 KB
orthography_difficult_triantafyllides_clue...	20/9/2021 12:33 μμ	Microsoft Excel W...	38 KB
photodentro_keywords_v1.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	13.853 KB
photodentro_keywords_with_wikipedia_i...	20/9/2021 12:33 μμ	Microsoft Excel W...	3.494 KB
photodentro_keywords_with_wikipedia_i...	20/9/2021 12:33 μμ	Microsoft Excel W...	3.387 KB
preposition_lexicon_lemma_v1.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	987 KB
quiz_questions_v1.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	35 KB
report.txt	20/9/2021 12:33 μμ	Text Document	185 KB
report_adjectives_tagged_in_photodentr...	20/9/2021 12:33 μμ	Microsoft Excel W...	409 KB
report_synonums_v1.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	32 KB
wikipedia_geography_subcategories.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	2.088 KB
wikipedia_terms_age_education_from_ph...	20/9/2021 12:33 μμ	Microsoft Excel W...	553 KB
xtizo_lekseis_prefixes_v1.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	2.721 KB
xtizo_lekseis_prefixes_word_b_lemma_ch...	20/9/2021 12:33 μμ	Microsoft Excel W...	0 KB
xtizo_lekseis_prefixes_word_semantic_clu...	20/9/2021 12:33 μμ	Microsoft Excel W...	24 KB
xtizo_lekseis_suffixes_v1.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	102 KB
xtizo_lekseis_suffixes_word_semantic_clu...	20/9/2021 12:33 μμ	Microsoft Excel W...	49 KB
ΚΩΔΙΚΟΠΟΙΗΣΗ ΛΑΘΩΝ_20_12-09.xls	20/9/2021 12:33 μμ	Microsoft Excel 97...	378 KB
ΟΡΘΟΓΡΑΦΙΑ_Συνηθισμένα λάθη.docx	20/9/2021 12:33 μμ	Microsoft Word D...	0 KB
περι_φωτόδεντρο_γεωγραφία_info_v1.x...	20/9/2021 12:33 μμ	Microsoft Excel W...	11 KB
υλικό_γεωγραφίας_γυμνασίου_v1.xlsx	20/9/2021 12:33 μμ	Microsoft Excel W...	413 KB

Επεξηγήσεις για το περιεχόμενο περιλαμβάνεται στο σχετικό αρχείο files_description_of_data_for_the_games_content.xlsx (βλέπε και επόμενη εικόνα).


files_description_of_data_for_the_games_content.xlsx - Microsoft Excel
A28
report_adjectives_tagged_in_photodentro_sentences_v3.xlsx
Όνομα Αρχείου
Περιγραφή
Τύπος
ebook Γεωγραφία
Φωτόδεντρο
ebook Λεξικό
Wikipedia
Επίθετα
Keyword
Χτίζω Λέξεις
Θεματική Ταξινομία
Μονολεκτικοί Όροι
Πολυλεκτικοί Όροι
Κύρια Ονόματα
Γραμματικά Φαινόμενα
Γραμματικός Διορθωτής
Word Embeddings
Αυτόματη Παραγωγή φράσεων
υλικό_γεωγραφίας_γυμνασίου_v1.xlsx
Τα κείμενα από τα μαθήματα της Γεωγραφία και όροι που έχουν βρεθεί με ορισμούς στα μαθήματα και επίσης έχουν βρεθει στην wikipedia
EXCEL YES YES YES YES
ΟΡΘΟΓΡΑΦΙΑ_Συνηθισμένα λάθη.docx
ΟΡΘΟΓΡΑΦΙΑ_Συνηθισμένα λάθη
DOC YES
wikipedia_terms_age_education_from_photodentro_keywords_v1.xlsx
Λέξεις κλειδιά από το Φωτόδεντρο που βρέθηκαν ακριβώς έτσι στην Wikipedia. Πληροφορία για βαθμίδα-ηλικία.
EXCEL YES YES
photodentro_keywords_with_wikipedia_info_same_v1.xlsx
Λέξεις κλειδιά από το Φωτόδεντρο που βρέθηκαν ακριβώς έτσι στην Wikipedia. Πληροφορία για βαθμίδα, ηλικία, θεματική περιοχή και υπόλοιπες λέξεις κλειδιά
EXCEL YES YES
photodentro_keywords_v1.xlsx
Λέξεις κλειδιά από το Φωτόδεντρο
EXCEL YES
photodentro_keywords_with_wikipedia_info_v1.xlsx
Λέξεις κλειδιά από το Φωτόδεντρο και πληροφορία απο το wikipedia όπου υπάρχει
EXCEL YES YES
ebooks_multiwords_with_wikipedia_info_v8.xlsx
Πολυλεκτικοί όροι από τα ebooks της Γεωγραφία με ένδειξη για κύρια ονόματα. Έχουν αναζητηθεί στην wikipedia και υπάρχει πληροφορία από αυτήν. Είναι συνδεδεμένοι με τα keywords και περιέχουν πληροφορία για την συχνότητα εμφάνισής τους στα ebooks.
EXCEL YES YES YES YES YES YES
ebooks_lemma_terms_with_wikipedia_info_v8.xlsx
μονολεκτικοί όροι από τα ebooks της Γεωγραφία με ένδειξη για κύρια ονόματα. Έχουν αναζητηθεί στην wikipedia και υπάρχει πληροφορία από αυτήν. Είναι συνδεδεμένοι με τα keywords και περιέχουν πληροφορία για την συχνότητα εμφάνισής τους στα ebooks.
EXCEL YES YES YES YES YES YES
ebooks_terms_lesson_category_v2.xlsx
Όροι από τα ebooks της Γεωγραφίας με πληροφορία για το μαθημα και κατηγοριοποιήση με βάση τη θεματική ταξινομία
EXCEL YES YES YES YES
wikipedia_geography_subcategories.xlsx
Διασύνδεση των κατηγοριών της Θεματικής Ταξινομίας (Γεωγραφία) και αυτών της Wikipedia
EXCEL YES YES
Lists_Templates_v2.xls
Λίστες με ορθογραφικά και συντακτικά λάθη. Έτοιμο υλικό για παιχνίδια ερωτήσεων ΣΩΣΤΟ/ΛΑΘΟΣ.
EXCEL YES YES
ΚΩΔΙΚΟΠΟΙΗΣΗ ΛΑΘΩΝ_20_12-09.xls
Λίστες με ορθογραφικά και συντακτικά λάθη. Έτοιμο υλικό για παιχνίδια ερωτήσεων ΣΩΣΤΟ/ΛΑΘΟΣ.
EXCEL YES YES
xtizo_lekseis_prefixes_word_b_lemma_check_v2_Feedback_All_V.1.4.xlsx
Χτίζω Λέξεις: λέξεις με α' συστατικό προθέσεις και διασταυρωμένα β' συστατικά
EXCEL YES
xtizo_lekseis_prefixes_v1.xlsx
Χτίζω Λέξεις: α' συστατικό - προθέσεις - περιγραφές - σχόλια
EXCEL YES
Sheet1
Ready

1.3. Αποτελέσματα Στατιστικής Επεξεργασίας Δεδομένων

Στο παράρτημα αυτό, καταγράφονται τα αρχεία που προέκυψαν από την επεξεργασία των δεδομένων με σκοπό να αποτελέσουν την είσοδο για τα παιχνίδια που υλοποιούνται. Τα ίδια τα αρχεία περιλαμβάνονται στο Παραδοτέο Π3.3, κάτω από το folder «Αποτελέσματα στατιστικής Επεξεργασίας Δεδομένων».



1.4. Βάση Δεδομένων

Στο παράρτημα αυτό δίνονται τα sql scripts με τις εντολές για την δημιουργία των πινάκων της βάσης δεδομένων στη MySQL που αναπτύχθηκε στα πλαίσια του για την αποθήκευση του χαρακτηρισμένου γλωσσικού υλικού. Επίσης υπάρχουν βοηθητικοί πίνακες οι οποίοι δημιουργήθηκαν για να καλύψουν τις ανάγκες των παιχνιδιών.

```
CREATE DATABASE  IF NOT EXISTS `lexipaignio_app` /*!40100 DEFAULT CHARACTER SET utf8 */;
USE `lexipaignio_app`;
-- MySQL dump 10.13  Distrib 5.6.10, for Win64 (x86_64)
--
-- Host: localhost    Database: lexipaignio_app
-- ------------------------------------------------------
-- Server version     5.6.10

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--
-- Table structure for table `clue`
--

DROP TABLE IF EXISTS `clue`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `clue` (
  `clue_id` bigint(20) NOT NULL AUTO_INCREMENT,
  `text` longtext NOT NULL,
  `text_normalized` longtext,
  `characters_count` int(11) DEFAULT NULL,
  PRIMARY KEY (`clue_id`)
) ENGINE=InnoDB AUTO_INCREMENT=4053 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `clue_properties`
--

DROP TABLE IF EXISTS `clue_properties`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `clue_properties` (
  `clue_id` bigint(20) NOT NULL,
  `property_id` bigint(20) NOT NULL,
  UNIQUE KEY `clue_property_ids_unique` (`clue_id`,`property_id`),
  KEY `FKjs4dnaey0674u4k0u0ooajund` (`property_id`),
```

```
  KEY `FKd04bcyekf9lv5mnpb7vnoh8pe` (`clue_id`),
  CONSTRAINT `FKd04bcyekf9lv5mnpb7vnoh8pe` FOREIGN KEY (`clue_id`) REFERENCES `clue` (`clue_id`),
  CONSTRAINT `FKjs4dnaey0674u4k0u0ooajund` FOREIGN KEY (`property_id`) REFERENCES `property` (`property_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `game`
--

DROP TABLE IF EXISTS `game`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `game` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `title` varchar(255) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `groups`
--

DROP TABLE IF EXISTS `groups`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `groups` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `active` bit(1) NOT NULL,
  `created_at` datetime(6) NOT NULL,
  `description` varchar(255) NOT NULL,
  `name` varchar(255) NOT NULL,
  `creator_id` bigint(20) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `FKjdwr03od1igxvnv8plt426cco` (`creator_id`),
  CONSTRAINT `FKjdwr03od1igxvnv8plt426cco` FOREIGN KEY (`creator_id`) REFERENCES `user` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_clue`
--

DROP TABLE IF EXISTS `instance_clue`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_clue` (
  `clue_id` bigint(20) NOT NULL AUTO_INCREMENT,
  `characters_count` int(11) NOT NULL,
  `text` longtext NOT NULL,
  `text_normalized` longtext,
  PRIMARY KEY (`clue_id`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_clue_properties`
```

```
--

DROP TABLE IF EXISTS `instance_clue_properties`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_clue_properties` (
  `clue_id` bigint(20) NOT NULL,
  `property_id` bigint(20) NOT NULL,
  PRIMARY KEY (`clue_id`,`property_id`),
  KEY `FK5l2ejbap75xys6em0h9i8myim` (`property_id`),
  CONSTRAINT `FK5l2ejbap75xys6em0h9i8myim` FOREIGN KEY (`property_id`) REFERENCES `instance_property` (`property_id`),
  CONSTRAINT `FKseoo6cnj7bc74a4qy8wu8noqx` FOREIGN KEY (`clue_id`) REFERENCES `instance_clue` (`clue_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_keys`
--

DROP TABLE IF EXISTS `instance_keys`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_keys` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `uq_key` varchar(12) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `UK_rk0ajw1vjcffomhi7vht9d0p2` (`uq_key`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_property`
--

DROP TABLE IF EXISTS `instance_property`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_property` (
  `property_id` bigint(20) NOT NULL AUTO_INCREMENT,
  `in_clue` tinyint(1) DEFAULT '0',
  `in_quiz` tinyint(1) DEFAULT '0',
  `in_term` tinyint(1) DEFAULT '0',
  `value` longtext NOT NULL,
  `mkey` varchar(190) NOT NULL,
  PRIMARY KEY (`property_id`),
  KEY `FKi26fsg1jl1djev40l8c2jdg73` (`mkey`),
  CONSTRAINT `FKi26fsg1jl1djev40l8c2jdg73` FOREIGN KEY (`mkey`) REFERENCES `property_keys` (`name`)
) ENGINE=InnoDB AUTO_INCREMENT=85 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_quiz`
--

DROP TABLE IF EXISTS `instance_quiz`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_quiz` (
```

```
  `quiz_id` bigint(20) NOT NULL AUTO_INCREMENT,
  `correct_answer_text` varchar(255) DEFAULT NULL,
  `quiz_question_id` bigint(20) NOT NULL,
  `correct_answer_term_id` bigint(20) NOT NULL,
  PRIMARY KEY (`quiz_id`),
  KEY `FKqtno8bcp9ypv8n9lhkyehdlwk` (`quiz_question_id`),
  KEY `FKr4vpesa85q3x9pux3xvmfhgnj` (`correct_answer_term_id`),
  CONSTRAINT `FKqtno8bcp9ypv8n9lhkyehdlwk` FOREIGN KEY (`quiz_question_id`) REFERENCES `instance_quiz_question` (`id`),
  CONSTRAINT `FKr4vpesa85q3x9pux3xvmfhgnj` FOREIGN KEY (`correct_answer_term_id`) REFERENCES `instance_term`
(`term_id`)
) ENGINE=InnoDB AUTO_INCREMENT=13 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_quiz_incorrect_answer`
--

DROP TABLE IF EXISTS `instance_quiz_incorrect_answer`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_quiz_incorrect_answer` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `text` varchar(255) DEFAULT NULL,
  `quiz_id` bigint(20) NOT NULL,
  `term_id` bigint(20) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `FK26xpnvxqbh968js0qyv0d7jh` (`quiz_id`),
  KEY `FKqwxefxkcfmelsng5452e7kosp` (`term_id`),
  CONSTRAINT `FK26xpnvxqbh968js0qyv0d7jh` FOREIGN KEY (`quiz_id`) REFERENCES `instance_quiz` (`quiz_id`),
  CONSTRAINT `FKqwxefxkcfmelsng5452e7kosp` FOREIGN KEY (`term_id`) REFERENCES `instance_term` (`term_id`)
) ENGINE=InnoDB AUTO_INCREMENT=55 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_quiz_properties`
--

DROP TABLE IF EXISTS `instance_quiz_properties`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_quiz_properties` (
  `quiz_id` bigint(20) NOT NULL,
  `property_id` bigint(20) NOT NULL,
  PRIMARY KEY (`quiz_id`,`property_id`),
  KEY `FKnfo5oegarayfga2etbf8j0b58` (`property_id`),
  CONSTRAINT `FKnfo5oegarayfga2etbf8j0b58` FOREIGN KEY (`property_id`) REFERENCES `instance_property` (`property_id`),
  CONSTRAINT `FKt8btem0d2afnxahw3i1nvt6s2` FOREIGN KEY (`quiz_id`) REFERENCES `instance_quiz` (`quiz_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_quiz_question`
--

DROP TABLE IF EXISTS `instance_quiz_question`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_quiz_question` (
```

```
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `characters_count` int(11) NOT NULL,
  `text` longtext NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=19 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_quiz_sets`
--

DROP TABLE IF EXISTS `instance_quiz_sets`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_quiz_sets` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `created_at` datetime(6) NOT NULL,
  `description` varchar(255) NOT NULL,
  `title` varchar(255) NOT NULL,
  `instance_key_id` bigint(20) NOT NULL,
  `user_id` bigint(20) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `FK6uukmmpdi4k105txfudcpv60p` (`instance_key_id`),
  KEY `FKobddcxkhysn4eyk6s9sr1kucv` (`user_id`),
  CONSTRAINT `FK6uukmmpdi4k105txfudcpv60p` FOREIGN KEY (`instance_key_id`) REFERENCES `instance_keys` (`id`),
  CONSTRAINT `FKobddcxkhysn4eyk6s9sr1kucv` FOREIGN KEY (`user_id`) REFERENCES `user` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_quiz_sets_groups`
--

DROP TABLE IF EXISTS `instance_quiz_sets_groups`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_quiz_sets_groups` (
  `quiz_set_id` bigint(20) NOT NULL,
  `group_id` bigint(20) NOT NULL,
  PRIMARY KEY (`quiz_set_id`,`group_id`),
  KEY `FKs3r76lcb9excexekfy3nauqgj` (`group_id`),
  CONSTRAINT `FKmoetuml4qmdkvs4xe49603yuo` FOREIGN KEY (`quiz_set_id`) REFERENCES `instance_quiz_sets` (`id`),
  CONSTRAINT `FKs3r76lcb9excexekfy3nauqgj` FOREIGN KEY (`group_id`) REFERENCES `groups` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_quiz_sets_quizzes`
--

DROP TABLE IF EXISTS `instance_quiz_sets_quizzes`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_quiz_sets_quizzes` (
  `quiz_set_id` bigint(20) NOT NULL,
  `quiz_id` bigint(20) NOT NULL,
  PRIMARY KEY (`quiz_set_id`,`quiz_id`),
  KEY `FKntt5ng09kyp3r5ewpjx0sbby3` (`quiz_id`),
```

```
  CONSTRAINT `FK4w9yxdkgghffabj3v6bm5dykd` FOREIGN KEY (`quiz_set_id`) REFERENCES `instance_quiz_sets` (`id`),
  CONSTRAINT `FKntt5ng09kyp3r5ewpjx0sbby3` FOREIGN KEY (`quiz_id`) REFERENCES `instance_quiz` (`quiz_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_term`
--

DROP TABLE IF EXISTS `instance_term`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_term` (
  `term_id` bigint(20) NOT NULL AUTO_INCREMENT,
  `characters_count` int(11) NOT NULL,
  `text` varchar(255) DEFAULT NULL,
  `text_normalized` varchar(255) DEFAULT NULL,
  `words_count` int(11) NOT NULL,
  PRIMARY KEY (`term_id`)
) ENGINE=InnoDB AUTO_INCREMENT=77 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_term_clues`
--

DROP TABLE IF EXISTS `instance_term_clues`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_term_clues` (
  `term_id` bigint(20) NOT NULL,
  `clue_id` bigint(20) NOT NULL,
  PRIMARY KEY (`term_id`,`clue_id`),
  KEY `FKsw2657b763h7k2vh5524kwtix` (`clue_id`),
  CONSTRAINT `FK18o1x7sk2a44l8o71l6gs2wbr` FOREIGN KEY (`term_id`) REFERENCES `instance_term` (`term_id`),
  CONSTRAINT `FKsw2657b763h7k2vh5524kwtix` FOREIGN KEY (`clue_id`) REFERENCES `instance_clue` (`clue_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_term_properties`
--

DROP TABLE IF EXISTS `instance_term_properties`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_term_properties` (
  `term_id` bigint(20) NOT NULL,
  `property_id` bigint(20) NOT NULL,
  PRIMARY KEY (`term_id`,`property_id`),
  KEY `FK7jdpdci5ljc9nqfsgrd4o0a06` (`property_id`),
  CONSTRAINT `FK7jdpdci5ljc9nqfsgrd4o0a06` FOREIGN KEY (`property_id`) REFERENCES `instance_property` (`property_id`),
  CONSTRAINT `FKkxh8wvn822hc4m5kqi9lw5sl6` FOREIGN KEY (`term_id`) REFERENCES `instance_term` (`term_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_term_sets`
```

```
--

DROP TABLE IF EXISTS `instance_term_sets`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_term_sets` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `created_at` datetime(6) NOT NULL,
  `description` varchar(255) NOT NULL,
  `title` varchar(255) NOT NULL,
  `instance_key_id` bigint(20) NOT NULL,
  `user_id` bigint(20) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `FKdkbor2hrspncacytaujnh5jq` (`instance_key_id`),
  KEY `FKnj6c110ilyptjw7o4d2brwhlj` (`user_id`),
  CONSTRAINT `FKdkbor2hrspncacytaujnh5jq` FOREIGN KEY (`instance_key_id`) REFERENCES `instance_keys` (`id`),
  CONSTRAINT `FKnj6c110ilyptjw7o4d2brwhlj` FOREIGN KEY (`user_id`) REFERENCES `user` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_term_sets_groups`
--

DROP TABLE IF EXISTS `instance_term_sets_groups`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_term_sets_groups` (
  `term_set_id` bigint(20) NOT NULL,
  `group_id` bigint(20) NOT NULL,
  PRIMARY KEY (`term_set_id`,`group_id`),
  KEY `FKr8pbftb4ifyijbu3giq1maqco` (`group_id`),
  CONSTRAINT `FK1fttrodojsa4u4uyd6g28wju9` FOREIGN KEY (`term_set_id`) REFERENCES `instance_term_sets` (`id`),
  CONSTRAINT `FKr8pbftb4ifyijbu3giq1maqco` FOREIGN KEY (`group_id`) REFERENCES `groups` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_term_sets_terms`
--

DROP TABLE IF EXISTS `instance_term_sets_terms`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_term_sets_terms` (
  `term_set_id` bigint(20) NOT NULL,
  `term_id` bigint(20) NOT NULL,
  PRIMARY KEY (`term_set_id`,`term_id`),
  KEY `FKetfuc7gd8t237clp7oplt08n` (`term_id`),
  CONSTRAINT `FKetfuc7gd8t237clp7oplt08n` FOREIGN KEY (`term_id`) REFERENCES `instance_term` (`term_id`),
  CONSTRAINT `FKi1c2iralrremyrbkneuqlb2dh` FOREIGN KEY (`term_set_id`) REFERENCES `instance_term_sets` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `instance_user_game_events`
--
```

```
DROP TABLE IF EXISTS `instance_user_game_events`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `instance_user_game_events` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `score` int(11) NOT NULL,
  `timestamp` datetime(6) NOT NULL,
  `game_id` bigint(20) NOT NULL,
  `instance_key_id` bigint(20) NOT NULL,
  `user_id` bigint(20) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `FKlk3fa2a6fumgaha4bgrrchgbh` (`game_id`),
  KEY `FKmv898s9bwopmghchunw5sf021` (`instance_key_id`),
  KEY `FK8a5vsrchrykdrug5lmv2cux9l` (`user_id`),
  CONSTRAINT `FK8a5vsrchrykdrug5lmv2cux9l` FOREIGN KEY (`user_id`) REFERENCES `user` (`id`),
  CONSTRAINT `FKlk3fa2a6fumgaha4bgrrchgbh` FOREIGN KEY (`game_id`) REFERENCES `game` (`id`),
  CONSTRAINT `FKmv898s9bwopmghchunw5sf021` FOREIGN KEY (`instance_key_id`) REFERENCES `instance_keys` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `property`
--

DROP TABLE IF EXISTS `property`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `property` (
  `property_id` bigint(20) NOT NULL AUTO_INCREMENT,
  `in_clue` tinyint(1) DEFAULT '0',
  `in_quiz` tinyint(1) DEFAULT '0',
  `in_term` tinyint(1) DEFAULT '0',
  `mkey` varchar(190) NOT NULL,
  `value` longtext NOT NULL,
  PRIMARY KEY (`property_id`),
  KEY `FKn27eyypqu7xqkljm1fyv1d2io` (`mkey`),
  CONSTRAINT `FKn27eyypqu7xqkljm1fyv1d2io` FOREIGN KEY (`mkey`) REFERENCES `property_keys` (`name`)
) ENGINE=InnoDB AUTO_INCREMENT=24977 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `property_keys`
--

DROP TABLE IF EXISTS `property_keys`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `property_keys` (
  `name` varchar(190) NOT NULL,
  `description` varchar(255) DEFAULT NULL,
  `label` varchar(255) DEFAULT NULL,
  `property_keyscol` bit(1) DEFAULT NULL,
  `property_keyscol1` bit(2) DEFAULT NULL,
  `property_keyscol2` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`name`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;
```

```
--
-- Table structure for table `quiz`
--

DROP TABLE IF EXISTS `quiz`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `quiz` (
  `quiz_id` bigint(20) NOT NULL AUTO_INCREMENT,
  `correct_answer_text` varchar(255) DEFAULT NULL,
  `quiz_question_id` bigint(20) NOT NULL,
  `correct_answer_term_id` bigint(20) NOT NULL,
  PRIMARY KEY (`quiz_id`),
  KEY `FK4vd596knsnpmnekit5iqd8vwe` (`quiz_question_id`),
  KEY `FKbw3ym2emnu5dyyewot2t8fufj` (`correct_answer_term_id`),
  CONSTRAINT `FK4vd596knsnpmnekit5iqd8vwe` FOREIGN KEY (`quiz_question_id`) REFERENCES `quiz_question` (`id`) ON
DELETE CASCADE,
  CONSTRAINT `FKbw3ym2emnu5dyyewot2t8fufj` FOREIGN KEY (`correct_answer_term_id`) REFERENCES `term` (`term_id`)
) ENGINE=InnoDB AUTO_INCREMENT=2823 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `quiz_incorrect_answer`
--

DROP TABLE IF EXISTS `quiz_incorrect_answer`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `quiz_incorrect_answer` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `text` varchar(255) DEFAULT NULL,
  `quiz_id` bigint(20) NOT NULL,
  `term_id` bigint(20) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `FKm9ne9wu1dbuoxxy1gl8avy7fx` (`quiz_id`),
  KEY `FK29k998fv4eukwe1o6ml8vivqx` (`term_id`),
  CONSTRAINT `FK29k998fv4eukwe1o6ml8vivqx` FOREIGN KEY (`term_id`) REFERENCES `term` (`term_id`),
  CONSTRAINT `FKm9ne9wu1dbuoxxy1gl8avy7fx` FOREIGN KEY (`quiz_id`) REFERENCES `quiz` (`quiz_id`) ON DELETE CASCADE
) ENGINE=InnoDB AUTO_INCREMENT=7690 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `quiz_properties`
--

DROP TABLE IF EXISTS `quiz_properties`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `quiz_properties` (
  `quiz_id` bigint(20) NOT NULL,
  `property_id` bigint(20) NOT NULL,
  UNIQUE KEY `quiz_property_ids_unique` (`quiz_id`,`property_id`),
  KEY `FKcne2a23dtoonnfimca0s1aleq` (`property_id`),
  KEY `FKnndh4hnd75238u7lhd1pblres` (`quiz_id`),
  CONSTRAINT `FKcne2a23dtoonnfimca0s1aleq` FOREIGN KEY (`property_id`) REFERENCES `property` (`property_id`),
  CONSTRAINT `FKnndh4hnd75238u7lhd1pblres` FOREIGN KEY (`quiz_id`) REFERENCES `quiz` (`quiz_id`) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;
```

```
--
-- Table structure for table `quiz_question`
--

DROP TABLE IF EXISTS `quiz_question`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `quiz_question` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `characters_count` int(11) NOT NULL,
  `text` longtext NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=2804 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `term`
--

DROP TABLE IF EXISTS `term`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `term` (
  `term_id` bigint(20) NOT NULL AUTO_INCREMENT,
  `characters_count` int(11) NOT NULL,
  `text_normalized` varchar(255) NOT NULL,
  `words_count` int(11) NOT NULL,
  `text` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`term_id`)
) ENGINE=InnoDB AUTO_INCREMENT=10670 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `term_clues`
--

DROP TABLE IF EXISTS `term_clues`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `term_clues` (
  `term_id` bigint(20) NOT NULL,
  `clue_id` bigint(20) NOT NULL,
  UNIQUE KEY `term_clue_ids_unique_index` (`clue_id`,`term_id`),
  KEY `FKb5620dgtmmmluddeey5nve2ll` (`clue_id`),
  KEY `FKtib8g52m2164atmlurodsu5aq` (`term_id`),
  CONSTRAINT `FKb5620dgtmmmluddeey5nve2ll` FOREIGN KEY (`clue_id`) REFERENCES `clue` (`clue_id`),
  CONSTRAINT `FKtib8g52m2164atmlurodsu5aq` FOREIGN KEY (`term_id`) REFERENCES `term` (`term_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `term_properties`
--

DROP TABLE IF EXISTS `term_properties`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
```

```
CREATE TABLE `term_properties` (
  `term_id` bigint(20) NOT NULL,
  `property_id` bigint(20) NOT NULL,
  UNIQUE KEY `term_property_ids_unique_index` (`term_id`,`property_id`),
  KEY `FKkiiqhprdjmtkdllg6q9fqg8ek` (`property_id`),
  KEY `FKekpct848vku65ndc81c5cr90e` (`term_id`),
  CONSTRAINT `FKekpct848vku65ndc81c5cr90e` FOREIGN KEY (`term_id`) REFERENCES `term` (`term_id`),
  CONSTRAINT `FKkiiqhprdjmtkdllg6q9fqg8ek` FOREIGN KEY (`property_id`) REFERENCES `property` (`property_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `user`
--

DROP TABLE IF EXISTS `user`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `user` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `created_at` datetime(6) NOT NULL,
  `enabled` bit(1) NOT NULL,
  `password` varchar(255) NOT NULL,
  `role` varchar(255) NOT NULL,
  `username` varchar(255) NOT NULL,
  `full_name` varchar(255) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=13 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `user_game_events`
--

DROP TABLE IF EXISTS `user_game_events`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `user_game_events` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `score` int(11) DEFAULT NULL,
  `timestamp` datetime(6) DEFAULT NULL,
  `game_id` bigint(20) NOT NULL,
  `user_id` bigint(20) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `FKk8fxoi50e0qw8lksl9w6wrm3g` (`game_id`),
  KEY `FKc9h17owrepqc0g1tid5r60rhn` (`user_id`),
  CONSTRAINT `FKc9h17owrepqc0g1tid5r60rhn` FOREIGN KEY (`user_id`) REFERENCES `user` (`id`),
  CONSTRAINT `FKk8fxoi50e0qw8lksl9w6wrm3g` FOREIGN KEY (`game_id`) REFERENCES `game` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=27 DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `user_groups`
--

DROP TABLE IF EXISTS `user_groups`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
```

```
CREATE TABLE `user_groups` (
  `group_id` bigint(20) NOT NULL,
  `user_id` bigint(20) NOT NULL,
  PRIMARY KEY (`group_id`,`user_id`),
  KEY `FKxgk67l5yp8458l39rog6nppe` (`user_id`),
  CONSTRAINT `FKmrgahbb4w32n9wkjqbipttc87` FOREIGN KEY (`group_id`) REFERENCES `groups` (`id`),
  CONSTRAINT `FKxgk67l5yp8458l39rog6nppe` FOREIGN KEY (`user_id`) REFERENCES `user` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping routines for database 'lexipaignio_app'
--
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2021-09-22  9:20:36
```

6. Κατηγορίες Λαθών Γραμματικού Διορθωτή

Α. ΥΦΟΛΟΓΙΑ

1. Έχουν κατηγοριοποιηθεί και περιγραφεί φράσεις της αρχαίας ελληνικής που διατηρούνται ακόμη στο γραπτό και προφορικό λόγο. Ο διορθωτής ενημερώνει το χρήστη (INFO) ότι, αν αυτές οι λέξεις συναντηθούν αυτόνομα στο λόγο, είναι αρχαιοπρεπείς. (pattern_1_ancient_phrase)
   ΚΑΤΗΓΟΡΙΕΣ

   - λέξεις με τελικό –ν (π.χ. ακατάλληλον….)
   - λέξεις με διατήρηση της ιστορικής ορθογραφίας –ιν, -ας, -ος, -ης, -ις, -εως. (π.χ. υπαγόρευσιν || αιώνας || αγώνος || ερεύνης || ακρόασις || πίστεως)
   - λέξεις που στη γενική ενικού διατηρούν τον τόνο στην παραλήγουσα (αληθείας)

2. Έχουν περιγραφεί ουσιαστικά της νέας ελληνικής που χρησιμοποιούνται στον προφορικό λόγο αλλά δεν προτιμώνται στο γραπτό. Ο διορθωτής ενημερώνει το χρήστη (INFO) ότι, αν αυτές τα ουσιαστικά λέξεις συναντηθούν αυτόνομα στο γραπτό λόγο, δεν είναι δόκιμες για το γραπτό (π.χ. δημαρχίνα || βδομάδα || μέθοδες || ξουράφι || Δεκέμβρη κ.λπ.). (pattern_2_oral_noun)

3. Έχουν κατηγοριοποιηθεί και περιγραφεί κλιτικοί τύποι που χρησιμοποιούνται στον προφορικό λόγο αλλά δεν προτιμώνται στο γραπτό. Ο διορθωτής ενημερώνει το χρήστη (INFO) ότι αυτοί οι τύποι δεν είναι δόκιμοι για το γραπτό λόγο (pattern_3_oral_verb).

   ΚΑΤΗΓΟΡΙΕΣ

   - τύποι ρημάτων συνηρημένων σε –άω (π.χ. αντανακλάω || αντανακλάει)
   - τύποι ρημάτων συνηρημένων σε –έω (π.χ. ασχολιέμαι)
   - τύποι ρήματος με κατάληξη –άνε || ούνε || -ουνα (π.χ. αγαπάνε || ασχολιόμουνα)
   - μεμονωμένοι τύποι ρημάτων (π.χ. ήσαντε, επέμβηκε)
   - φράσεις που δεν είναι δόκιμες στο γραπτό λόγο (π.χ. είσαι λάθος)
   - παραγωγικά επιθήματα που χρησιμοποιούνται στον προφορικό λόγο (-απόκτησα || διάνυσα)
   - λήμματα ρημάτων με υφολογικό χαρακτηρισμό: προφορικό (π.χ. θρέφω || ξομολογάω || ξαποστέλνω)

4. Έχουν περιγραφεί επίθετα της νέας ελληνικής που χρησιμοποιούνται στον προφορικό λόγο αλλά δεν προτιμώνται στο γραπτό. Ο διορθωτής ενημερώνει το χρήστη (INFO) ότι δεν είναι δόκιμες για το γραπτό (π.χ. δημαρχίνα || βδομάδα || μέθοδες || ξουράφι || Δεκέμβρη κ.λπ.). (pattern_4_oral_adj)

   ΚΑΤΗΓΟΡΙΕΣ

   - τύποι επιθέτων (ΘΗΛΥΚΟ) με παραγωγικό επίθημα –ούχα. Στην προκειμένη περίπτωση ελέγχεται η συμφωνία υποκειμένου κατηγορουμένου ή η συμφωνία επιθέτου ουσιαστικού μέσα σε ονοματική φράση. (π.χ. η πορτοκαλάδα είναι αεριούχα || η αριστούχα μαθήτρια).
   - Λήμματα επιθέτων που χαρακτηρίζονται ως προφορικά (π.χ. γιομάτος || αψηλός κ.λπ.)

5. Έχουν περιγραφεί συγκεκριμένοι μορφολογικοί τύποι της μετοχής που, σε συγκεκριμένη ονοματική φράση, χρησιμοποιούνται στον προφορικό λόγο αλλά δεν προτιμώνται στο γραπτό (π.χ. διαταγμένη υπηρεσία). Ο διορθωτής ενημερώνει το χρήστη (INFO) ότι αυτοί οι τύποι δεν είναι δόκιμοι για το γραπτό λόγο (pattern_5_oral_part).

6. Έχουν κατηγοριοποιηθεί συγκεκριμένα παραγωγικά επιθήματα παρελθοντικών χρόνων που είναι κατάλοιπα της αρχαίας ελληνικής ή της νέας ελληνικής και δε χρησιμοποιούνται στον λόγο (π.χ. θεωρείτο || προηγούσαντε). Οι μορφολογικοί αυτοί τύποι δεν είναι περιγραμμένοι στο λεξικό, γι' αυτό και επιλέγεται η ανάκληση τους μέσω των παραγωγικών επιθεμάτων. Ο διορθωτής ενημερώνει το χρήστη (INFO) ότι αυτοί οι τύποι δεν είναι δόκιμοι για το γραπτό λόγο (pattern_6_oral_ancient_past_tense).

7. Έχουν περιγραφεί συγκεκριμένες φράσεις της νέας ελληνικής που δε χρησιμοποιούνται στο γραπτό λόγο (π.χ. έχει ύπαρξη ζωής). Ο διορθωτής ενημερώνει το χρήστη (INFO) ότι αυτοί οι τύποι δεν είναι δόκιμοι για το γραπτό λόγο. Στη συγκεκριμένη κατηγορία θα περιγραφούν και όλοι οι μορφολογικοί τύποι που έχουν από το μορφολογικό λεξικό το attribute ή TAG: INFORMAL, ORAL (pattern_7_oral_speech_general).

8. Έχουν περιγραφεί συγκεκριμένα μεμονωμένα λήμματα της αρχαίας ελληνικής που καλό είναι να μην χρησιμοποιούνται στο γραπτό λόγο (π.χ. ευτυχής || ευώδης || ευειδής). Ο διορθωτής ενημερώνει το χρήστη (INFO) ότι αυτοί οι τύποι δεν είναι δόκιμοι για το γραπτό νεοελληνικό λόγο. (pattern_33_ancient_word).

9. Έχουν περιγραφεί συγκεκριμένα συμφωνικά συμπλέγματα (παραγωγικά επιθήματα, προθήματα) που είτε χρησιμοποιούνται στον προφορικό λόγο είτε είναι λανθασμένα (π.χ. απόχτησα || εννιά). Ο διορθωτής ενημερώνει το χρήστη (INFO ή WRONG) ότι αυτοί οι τύποι δεν είναι δόκιμοι για το γραπτό νεοελληνικό λόγο. (pattern_55_wrong_oral_consonant).

10. Έχουν περιγραφεί επιρρήματα που χρησιμοποιούνται ευρέως στον προφορικό λόγο (π.χ. 'τουλάχιστο' αντί του ορθού 'τουλάχιστον'). Ο διορθωτής ενημερώνει το χρήστη (INFO) ότι αυτοί οι τύποι δεν είναι δόκιμοι για το γραπτό νεοελληνικό λόγο. (pattern_56_oral_adv).

Β. ΣΗΜΕΙΑ ΣΤΙΞΗΣ

ΑΠΟΣΤΡΟΦΟΣ

11. Έχουν περιγραφεί οι κανόνες για προσθήκη ή αφαίρεση αποστρόφου. Ο χρήστης ενημερώνεται (WRONG) αν πρέπει να αφαιρέσει ή να προσθέσει την απόστροφο (pattern_9_punctuation_apostr) (π.χ. εξ' ύψους ||καθ οδόν). Στο συγκεκριμένο pattern δεν υπάρχει γενικευμένη χρήση κανόνων, αλλά έχουν ληφθεί υπόψη συγκεκριμένες προθετικές φράσεις στις οποίες η απόστροφος χρησιμοποιείται ενώ οι ίδιες προθέσεις σε άλλα περιβάλλοντα έχουν διαφορετική συμπεριφορά (π.χ. η πρόθεση από). Στη συγκεκριμένη κατηγορία περιγράφονται και α) λέξεις που η προσθήκη αποστρόφου απαιτεί και το χωρισμό τους σε δύο όπως επίσης και β) η συμπεριφορά του συνδέσμου (και -κι) σε συγκεκριμένα περιβάλλοντα λέξεων.

ΤΟΝΟΣ

12. Έχουν περιγραφεί οι κανόνες για προσθήκη ή αφαίρεση τόνου σε συγκεκριμένες λέξεις. Ο χρήστης ενημερώνεται (WRONG) αν πρέπει να αφαιρέσει ή να προσθέσει τον τόνο (pattern_10_punctuation_stress) (π.χ. κύρ – Δημήτρης || από που και ως που || θα μπω). Επιπλέον υπάρχουν και συγκεκριμένες ονοματικές φράσεις που γράφονται με λανθασμένο τόνο (π.χ.: οδός Ατθιδών)

ΕΝΩΤΙΚΟ

13. Έχουν περιγραφεί οι κανόνες για προσθήκη ή αφαίρεση παύλας σε συγκεκριμένες λέξεις. Ο χρήστης ενημερώνεται (WRONG) αν πρέπει να αφαιρέσει ή να προσθέσει την παύλα (pattern_11_punctuation_hyphen) (π.χ. μπαρμπα Νίκος || κάτω - κάτω). Επιπλέον υπάρχουν και συγκεκριμένες ονοματικές φράσεις που γράφονται λανθασμένα με ή χωρίς ενωτικό (π.χ.: παιδί θαύμα ||πάνω - κάτω)

ΚΟΜΜΑ

14. Έχουν περιγραφεί οι κανόνες για προσθήκη ή αφαίρεση κόμματος σε συγκεκριμένες λέξεις. Ο χρήστης ενημερώνεται (WRONG) αν πρέπει να αφαιρέσει ή να προσθέσει την παύλα (pattern_12_punctuation_komma) (π.χ. ο,τιδήποτε || έρχομαι επειδή).

ΤΕΛΕΙΑ

15. Έχουν περιγραφεί οι κανόνες για προσθήκη ή αφαίρεση τελείας σε συγκεκριμένο γλωσσικό περιβάλλον (αφαίρεση μετά από ερωτηματικό, προσθήκη μετά την παρένθεση κ.λπ.). Ο χρήστης ενημερώνεται (WRONG) αν πρέπει να αφαιρέσει ή να προσθέσει την τελεία (pattern_13_punctuation_dot).

Γ. ΛΑΝΘΑΣΜΕΝΗ ΓΡΑΦΗ ΦΡΑΣΕΩΝ

ΟΡΘΟΓΡΑΦΙΑ

16. Έχουν περιγραφεί ονοματικές φράσεις ευρείας χρήσης που υπάρχει λανθασμένη ορθογραφία ή επιλογή λέξεων. (π.χ. ψηλή κυριότητα). Ο χρήστης ενημερώνεται (WRONG) ότι η επιλογή του επιθέτου ψηλή στο συγκεκριμένο περιβάλλον είναι λανθασμένη και ως εκ τούτου πρέπει να επιλέξει το ψιλή (pattern_14_wrong_spelling).

ΓΡΑΦΗ

17. Έχουν περιγραφεί συγκεκριμένες ονοματικές ή ρηματικές φράσεις ή ρήματα ευρείας χρήσης που υπάρχει λανθασμένη γραφή. (π.χ. αντεπεξέρχομαι || προτού έρθεις || κυρίας Βουγιουκλάκης). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της συγκεκριμένης λέξης είναι λανθασμένη (μορφολογικά ή συντακτικά και προτείνεται η σωστός τύπος ή γραφή (pattern_15_wrong_write_phrase).

ΓΡΑΦΗ (1 ΛΕΞΗ)

18. Έχουν περιγραφεί συγκεκριμένες φράσεις (ελληνικής ή άλλης προέλευσης) που λανθασμένα γράφονται σε 2 λέξεις αντί μιας. (π.χ. επί κεφαλής || ες αεί κ.λπ.). Ο χρήστης ενημερώνεται

(WRONG) ότι ο σχηματισμός της φράσης είναι λανθασμένος και προτείνει τη γραφή της φράσης σε μία λέξη (pattern_16_one_word).

ΓΡΑΦΗ (2 ΛΕΞΕΙΣ)

19. Έχουν περιγραφεί συγκεκριμένες φράσεις (ελληνικής ή άλλης προέλευσης) που λανθασμένα γράφονται σε 1 λέξη αντί δύο. (π.χ. καταναλογία || μπάσκετμπολ κ.λπ.). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της φράσης είναι λανθασμένος και προτείνει τη γραφή της λέξης σε δύο λέξεις (pattern_17_two_words).

Δ. ΓΡΑΜΜΑΤΙΚΑ ΛΑΘΗ

ΤΕΛΙΚΟ -Ν

20. Έχουν περιγραφεί οι κανόνες του τελικού -ν. Οι κανόνες κατηγοριοποιούνται σε 2 ευρείες κατηγορίες (προσθήκη του τελικού -ν: 15 κανόνες και αφαίρεση του τελικού -ν: 20 κανόνες). Ο χρήστης ενημερώνεται (WRONG) αν πρέπει να αφαιρέσει ή να προσθέσει το τελικό –ν (pattern_8_teliko_n). Επειδή στην κατηγορία αυτή υπάρχει έντονη αμφισημία και διαφορετική συμπεριφορά του οριστικού άρθρου και της προσωπικής αντωνυμίας, χρησιμοποιείται κυρίως σε αυτό το pattern ο tagger (55 κανόνες).

ΕΠΑΝΑΛΗΨΗ

21. Έχουν περιγραφεί φράσεις (ονοματικές, προθετικές, ρηματικές) στις οποίες υπάρχει πλεονασμός κάποιου στοιχείου. Η κατηγορία αυτή περιλαμβάνει συγκεκριμένα λάθη και φράσεις (π.χ. από ανέκαθεν) όπως επίσης και γενικότερα λανθασμένο σχηματισμό φράσης (πιο καλύτερος = πιο + συγκριτικός). Ο χρήστης ενημερώνεται (WRONG) ότι στη συγκεκριμένη φράση υπάρχει πλεονασμός και προτείνει την απαλοιφή λέξης ή (pattern_18_redundancy).

ΠΛΗΘΥΝΤΙΚΟΣ ΑΡΙΘΜΟΣ

22. Έχουν περιγραφεί λέξεις στις οποίες υπάρχει λανθασμένος σχηματισμός πληθυντικού. (π.χ. ζωές || ανθρωπότητες). Ο χρήστης ενημερώνεται (WRONG) ότι ο πληθυντικός του συγκεκριμένου λήμματος δε χρησιμοποιείται στον επίσημο γραπτό λόγο (pattern_19_no_plural). Στην κατηγορία αυτή έχουν αποκλειστεί συγκεκριμένες φράσεις που ο πληθυντικός του λήμματος θεωρείται ορθός (π.χ. έχει εφτά ζ**ωές**).

23. Έχουν περιγραφεί λέξεις στις οποίες υπάρχει λανθασμένος σχηματισμός γενικής πληθυντικού. (π.χ. ζαχαρών || μπλουζών). Ο χρήστης ενημερώνεται (WRONG) ότι η γενική πληθυντικού του συγκεκριμένου λήμματος δεν είναι εύχρηστη στον επίσημο γραπτό λόγο (pattern_48_no_gen_plur) και προτείνεται η αντικατάστασή της με εμπρόθετο προσδιορισμό.

ΠΑΡΕΛΘΟΝΤΙΚΟΙ ΧΡΟΝΟΙ

24. Έχουν περιγραφεί συγκεκριμένοι τύποι ρημάτων στους οποίους υπάρχει λανθασμένος σχηματισμός παρελθοντικού χρόνου. (π.χ. έχει απηυδήσει || έχουν ειδωθεί). Στην κατηγορία αυτή επιπρόσθετα περιγράφονται και ομάδες ρημάτων που σχηματίζουν λανθασμένα τους παρελθοντικούς χρόνους (π.χ. τα σύνθετα του βάλλω, άγω, τείνω). Ο χρήστης ενημερώνεται

(WRONG) ότι ο σχηματισμός του συγκεκριμένου χρόνου (περιφραστικού ή μονολεκτικού) είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός (pattern_20_wrong_past_tense).

ΜΕΛΛΟΝΤΑΣ

25. Έχουν περιγραφεί συγκεκριμένοι τύποι ρημάτων στους οποίους υπάρχει λανθασμένος σχηματισμός μέλλοντα. (π.χ. θα απονείμει). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός του μέλλοντα είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός (pattern_21_wrong_future).

ΛΑΝΘΑΣΜΕΝΗ ΚΑΤΑΛΗΞΗ ΜΕΤΟΧΗΣ

26. Στη συγκεκριμένη κατηγορία έχει περιγραφεί η λανθασμένη κατάληξη (τονικά ή μορφολογικά) αρχαιόκλιτων – αλλά σε χρήση στα νέα ελληνικά – μετοχών (π.χ. ‘το αναλογόν ποσό’ αντί του ορθού ‘το αναλογούν ποσό’ || ‘οι συμμετάσχοντες’ αντί του ορθού ‘συμμετασχόντες’). Ο χρήστης ενημερώνεται (WRONG) ότι ο μορφολογικός σχηματισμός της μετοχής είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της (pattern_30_wrong_suffix_metoxi).

ΛΑΝΘΑΣΜΕΝΟΣ ΣΧΗΜΑΤΙΣΜΟΣ ΧΡΟΝΟΥ

27. Στη συγκεκριμένη κατηγορία έχουν περιγραφεί μεμονωμένοι τύποι – εσφαλμένοι γραμματικά – που έχουν ενταχθεί στην καθημερινή προφορική νόρμα αλλά όμως δεν πρέπει να χρησιμοποιούνται στον επίσημο γραπτό λόγο (π.χ. ‘τίθονται’ αντί του ορθού ‘τίθενται’). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός του συγκεκριμένου τύπου είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός του (pattern_32_wrong_tense).

ΛΑΝΘΑΣΜΕΝΟΣ ΣΧΗΜΑΤΙΣΜΟΣ ΦΩΝΗΣ

28. Στη συγκεκριμένη κατηγορία έχουν περιγραφεί τύποι ρημάτων που εσφαλμένα χρησιμοποιούνται στην ενεργητική ή μέση φωνή όπως επίσης και η λανθασμένη σύνταξη τους (π.χ. ‘συνεπάγει’ αντί του ορθού ‘συνεπάγεται’ || ‘ελλοχεύει τον κίνδυνο’ αντί του ορθού ‘ελλοχεύει τον κίνδυνο’. Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός του συγκεκριμένου τύπου είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός του (pattern_34_wrong_voice).

ΛΑΝΘΑΣΜΕΝΟΣ ΣΧΗΜΑΤΙΣΜΟΣ ΕΓΚΛΙΣΗΣ

29. Στη συγκεκριμένη κατηγορία έχει περιγραφεί ο λανθασμένος ή ο προφορικός σχηματισμός έγκλισης τύπων ρημάτων (π.χ. ‘τηλεφωνείστε’ αντί του ορθού ‘τηλεφωνήστε’ || ‘να βάλλεις’ αντί του ορθού ‘να βάλεις’). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της συγκεκριμένης έγκλισης είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της (pattern_35_wrong_egglisi).

ΛΑΝΘΑΣΜΕΝΗ ΓΡΑΦΗ ΛΟΓΙΑΣ ΦΡΑΣΗΣ

30. Στη συγκεκριμένη κατηγορία έχουν αποδελτιωθεί οι πιο συχνά χρησιμοποιούμενες λόγιες φράσεις όπως επίσης και τα πιο συχνά λάθη στη γραφή τους (ορθογραφικά λάθη: π.χ. ‘αιδός Αργείοι’ αντί του ορθού ‘αιδώς Αργείοι’ , συντακτικά: π.χ. ‘αυτής καθ’ εαυτής’ αντί του ορθού ‘αυτής καθ’ εαυτήν’ || λεξιλογικά: π.χ. ‘εξ ιδίων τα βέλη’ αντί του ορθού ‘εξ οικείων τα βέλη’). Ο χρήστης

ενημερώνεται (WARNING) ότι ο σχηματισμός της συγκεκριμένης λόγιας φράσης είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της (pattern_36_corpus_ancient_phrase).

Δ. ΣΥΝΤΑΚΤΙΚΑ ΛΑΘΗ

ΣΥΜΦΩΝΙΑ ΑΡΘΡΟΥ - ΟΥΣΙΑΣΤΙΚΟΥ

31. Στη συγκεκριμένη κατηγορία έχουν περιγραφεί προβλήματα συμφωνίας άρθρου ουσιαστικού. Η συμφωνία μπορεί να περιορίζεται στο γένος (π.χ. του περιπόλου), στον αριθμό (π.χ. των μία και μισή) και στην παγιωμένη αντίληψης περί ακλισίας κάποιων τύπων (π.χ. τον πάτερ). Επιπλέον η κατηγορία αυτή επεκτείνεται και σε ευρύτερη κατηγορία λαθών όπως τα θηλυκά σε –ος που λογίζονται στον καθημερινό λόγο ως αρσενικά. (π.χ. ο ψήφος || τους μεθόδους). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της συγκεκριμένης ονοματικής φράσης είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της (pattern_22_agreement_art_noun).

ΣΥΜΦΩΝΙΑ ΑΡΘΡΟΥ - ΕΠΙΘΕΤΟΥ

32. Στη συγκεκριμένη κατηγορία έχουν περιγραφεί προβλήματα συμφωνίας άρθρου - επιθέτου. Ο λανθασμένος σχηματισμός βρίσκεται στο γένος. Στη συγκεκριμένη κατηγορία περιλαμβάνονται ευρύτερες κατηγορίες επιθέτων (επίθετα σε –ύς, -εία, -ύ) στα οποία υπάρχει λανθασμένη επιλογή του αρσενικού ή του θηλυκού γένους (π.χ. των παχέων γυναικών || των βαθειών ανθρώπων). Επιπλέον η κατηγορία αυτή επεκτείνεται και σε ευρύτερη κατηγορία λαθών όπως τα θηλυκά σε –ος που σε ευρύτερες ονοματικές φράσεις λογίζονται λόγο ως αρσενικά. (π.χ. αυτοί οι μέθοδοι || αυτοί οι μέθοδοιΟ χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της συγκεκριμένης ονοματικής φράσης είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της (pattern_23_agreement_art_adj).

ΣΥΜΦΩΝΙΑ ΜΕΤΟΧΗΣ – ΟΥΣΙΑΣΤΙΚΟΥ & ΕΠΙΘΕΤΟΥ ΟΥΣΙΑΣΤΙΚΟΥ

33. Στη συγκεκριμένη κατηγορία έχουν περιγραφεί προβλήματα συμφωνίας μετοχής - ουσιαστικού. Ο λανθασμένος σχηματισμός βρίσκεται στο γένος. Στη συγκεκριμένη κατηγορία περιλαμβάνονται συγκεκριμένες μετοχές που η γενική του αρσενικού χρησιμοποιείται και στα θηλυκά (π.χ. πληγέντων περιοχών || δευτερεύοντος σημασίας). Επιπλέον η κατηγορία αυτή επεκτείνεται και στη συμφωνία επιθέτου ουσιαστικού και συγκεκριμένα στις κατηγορίες: α) πολλή + επίθετο αντί του ορθού πολύ + επίθετο, β) πολύ + ουσιαστικό αντί του ορθού πολλή + ουσιαστικό, γ) τα επίθετα σε –ης, -ης, -ες (π.χ. διαμπερή διαμέρισμα, ευγενής παιδί) δ) στο λανθασμένο σχηματισμών του επιθέτου που προσδιορίζει τη λέξη ετών (π.χ. ενενήντα τρία ετών), ε) στη λανθασμένη χρήση των επιθέτων ‘ανεξαρτήτου’ και ‘αδιακρίτου’ σε συνδυασμό με γενική θηλυκού και στ) στη λανθασμένη χρήση του επιθέτου ‘πολλοί’ σε συνδυασμό με πληθυντικό αρσενικού (πολλοί καλοί, αντί του ορθού πολύ καλοί). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της συγκεκριμένης ονοματικής φράσης είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της (pattern_24_agreement_part_adj_noun).

ΣΥΜΦΩΝΙΑ ΟΥΣΙΑΣΤΙΚΟΥ – ΟΥΣΙΑΣΤΙΚΟΥ

34. Στη συγκεκριμένη κατηγορία έχουν περιγραφεί προβλήματα συμφωνίας ουσιαστικού - ουσιαστικού. Υπάρχουν ονοματικές φράσεις που το συμπλήρωμα α) σε γενική ($2^{η}$ λέξη) είναι λανθασμένο και χρειάζεται να αντικατασταθεί από προθετική φράση ή ονομαστική (π.χ. πλήθος πουλιών αντί πλήθος από πουλιά || 15 ώρες γυμναστικής αντί του ορθού 15 ώρες γυμναστική) ή β)

σε ονομαστική χρειάζεται να αντικατασταθεί από ονομαστική (π.χ. ύπαρξη πλαγκτόν αντί του ορθού ύπαρξη πλαγκτού). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της συγκεκριμένης ονοματικής φράσης είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της (pattern_25_agreement_noun_noun).

ΣΥΜΦΩΝΙΑ ΕΠΙΡΡΗΜΑΤΙΚΗΣ ΦΡΑΣΗΣ

35. Στη συγκεκριμένη κατηγορία έχουν περιγραφεί προβλήματα λανθασμένης γραφής επιρρηματικής φράσης: α) λανθασμένη χρήση πτώσης (π.χ. από θέσης ισχύος) β) ελλιπής σχηματισμός της επιρρηματικής φράσης (π.χ. εκτός ότι αντί του ορθού εκτός του ότι) γ) λανθασμένος ορθογραφικός σχηματισμός των δοτικοφανών επιρρημάτων βάσει, φύσει δ) λανθασμένη επιλογή επιρρήματος σε συγκεκριμένο περιβάλλον (π.χ. παραβιάζω παράφορα αντί του ορθού παραβιάζω καταφορά). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της συγκεκριμένης επιρρηματικής φράσης είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της(pattern_26_agreement_adv_phrase).

ΣΥΜΦΩΝΙΑ ΕΥΡΕΙΑΣ ΟΝΟΜΑΤΙΚΗΣ ΦΡΑΣΗΣ

36. Στη συγκεκριμένη κατηγορία έχουν περιγραφεί 12 φράσεις με λανθασμένη επιλογή λέξης σε συγκεκριμένο περιβάλλον λέξεων (π.χ. ‘ραγδαία βελτίωση’ αντί του ορθού ‘θεαματική βελτίωση’). Στην κατηγορία αυτή δεν εξετάζεται μόνο η συμφωνία επιθέτου ουσιαστικού, όταν οι δύο λέξεις είναι όμορες αλλά και σε απομακρυσμένο περιβάλλον όπως επίσης και η συμφωνία υποκειμένου κατηγορουμένου. Στη συγκεκριμένη κατηγορία περιγράφονται και προβλήματα με λανθασμένη πτώση αντωνυμίας (τόσοι πολλοί αντί του ορθού τόσο πολλοί || καθένας άνθρωπος αντί του ορθού κάθε άνθρωπος), όπως επίσης και λανθασμένη χρήση ευρύτερης φράσης με επίθετα σε –ης (π.χ. επίκειται διεθνή συμφωνία) αλλά και προβλήματα λανθασμένης ορθογραφίας της γενικής των επιθέτων σε –υς που συγχέεται με το επίρρημα (π.χ. του ευρέως συνασπισμού αντί του ορθού ευρέος συνασπισμού) Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της συγκεκριμένης ονομαστικής φράσης είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της(pattern_27_agreement_nphrase).

ΣΥΜΦΩΝΙΑ ΡΗΜΑΤΙΚΗΣ ΦΡΑΣΗΣ

37. Στη συγκεκριμένη κατηγορία έχει περιγραφεί η λανθασμένη πτώση υποκειμένου σε ρήματα (π.χ. ‘ευχαριστούμε όλους **όσους** ήρθαν’ αντί του ορθού ‘ευχαριστούμε όλους **όσοι** ήρθαν’). Στο ίδιο pattern περιγράφονται και ρήματα που στη νέα ελληνική δεν πρέπει να διατηρούν στην αρχαιόκλιτη σύνταξη τους (π.χ. αποποιούμαι της ευθύνης) όπως επίσης και ρήματα που χρησιμοποιούνται λανθασμένα σε συγκεκριμένο γλωσσικό περιβάλλον (π.χ. ‘εκτίω ποινή’ αντί του ορθού ‘εκτίνω ποινή’). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της συγκεκριμένης ρηματικής φράσης είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της (pattern_28_agreement_vphrase).

ΣΥΜΦΩΝΙΑ ΠΡΟΘΕΤΙΚΗΣ ΦΡΑΣΗΣ

38. Στη συγκεκριμένη κατηγορία έχει περιγραφεί ο λανθασμένος σχηματισμός προθετικής φράσης (π.χ. ‘ενάντιων όλων των …’ αντί του ορθού ‘εναντίον όλων των ….’ || ‘με ότι’ αντί του ορθού ‘με ό,τι’). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της συγκεκριμένης προθετικής φράσης είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της (pattern_29_agreement_pre_phrase).

ΣΥΜΦΩΝΙΑ ΑΡΘΡΟΥ - ΑΝΤΩΝΥΜΙΑΣ

39. Στη συγκεκριμένη κατηγορία έχει περιγραφεί η περιττή χρήση του άρθρου σε συγκεκριμένα γλωσσικά περιβάλλοντα με αντωνυμίες. (π.χ. ‘ο οποιοσδήποτε’ αντί του ορθού ‘οποιοσδήποτε’). Ο χρήστης ενημερώνεται (WRONG) ότι ο σχηματισμός της συγκεκριμένης φράσης είναι λανθασμένος και προτείνεται ο ορθός σχηματισμός της (pattern_31_agreement_art_pron).

Ε. ΣΗΜΑΣΙΟΛΟΓΙΑ

Στην κατηγορία αυτή κατατάσσονται όλα τα σημασιολογικά ζητήματα, δηλαδή ζητήματα εννοιολογικής ή/και ορθογραφικής σύγχυσης καθώς και λεξιλογικών ισοδυναμιών.

40. Έχουν κατηγοριοποιηθεί και περιγραφεί επίθετα, επιρρήματα ουσιαστικά, ρήματα και φράσεις που συγχέονται εννοιολογικά (π.χ. απλά και απλώς). Στην κατηγορία αυτή υπάρχουν 2 επίπεδα επεξεργασίας. Σε πρώτο επίπεδο έχει καταγραφεί το γλωσσικό περιβάλλον με το οποίο συνδέονται τα συγκεκριμένα λήμματα. Ως εκ τούτου ο χρήστης δεν ενημερώνεται (INFO) για πιθανή εννοιολογική σύγχυση του συγκεκριμένου λήμματος, δεδομένου ότι η επιλογή του στο συγκεκριμένο γλωσσικό περιβάλλον είναι ορθή. Σε δεύτερο επίπεδο ο χρήστης ενημερώνεται για πιθανή εννοιολογική σύγχυση του χρησιμοποιούμενου λήμματος, ενώ παράλληλα δίνεται και η ερμηνεία και των 2 λημμάτων. (pattern_37_samepron_adj || pattern_38_samepron_adv || pattern_39_samepron_noun || pattern_40_samepron_verb)

41. Έχουν κατηγοριοποιηθεί και περιγραφεί επίθετα, επιρρήματα ουσιαστικά, ρήματα και φράσεις που είναι ομόηχα με ελάχιστες ορθογραφικές διαφορές και τα οποία συγχέονται εννοιολογικά (π.χ. φύλλο και φύλο || τοίχος και τείχος). Στην κατηγορία αυτή υπάρχουν 2 επίπεδα επεξεργασίας. Σε πρώτο επίπεδο έχει καταγραφεί το γλωσσικό περιβάλλον με το οποίο συνδέονται τα συγκεκριμένα λήμματα. Ως εκ τούτου ο χρήστης δεν ενημερώνεται (INFO) για πιθανή εννοιολογική σύγχυση του συγκεκριμένου λήμματος, δεδομένου ότι η επιλογή του στο συγκεκριμένο γλωσσικό περιβάλλον είναι ορθή. Σε δεύτερο επίπεδο ο χρήστης ενημερώνεται για πιθανή εννοιολογική σύγχυση του χρησιμοποιούμενου λήμματος, ενώ παράλληλα δίνεται και η ερμηνεία και των 2 λημμάτων. (pattern_41_omopron_words)

42. Έχουν κατηγοριοποιηθεί και περιγραφεί ξένες λέξεις ή φράσεις που δεν έχουν ενταχθεί στο κλιτικό σύστημα της νέας ελληνικής και για τις οποίες υπάρχει αντίστοιχη ελληνική λέξη (π.χ. μάσκετ μπολ). Ο χρήστης ενημερώνεται (INFO) για το γεγονός ότι η λέξη έχει ξενική προέλευση και του δίνεται η η δυνατότητα να επιλέξει την αντίστοιχη ελληνική. (pattern_42_ksenes_lexeis)

43. Έχουν κατηγοριοποιηθεί και περιγραφεί λατινικές λέξεις ή φράσεις και για τις οποίες υπάρχει αντίστοιχη ελληνική λέξη (π.χ. a posteriori). Ο χρήστης ενημερώνεται (INFO) για την αντίστοιχη ερμηνεία της λατινικής και του δίνεται η η δυνατότητα να επιλέξει την αντίστοιχη ελληνική. (pattern_49_Latin_phrases)

7. Βιβλιογραφία

P. Gakis, C. Panagiotakopoulos, K. Sgarbas & C. Tsalidis, Design and implementation of an electronic lexicon for Modern Greek, Literary and Linguistic Computing, Volume 27, Issue 2, June 2012, Pages 155–169, https://doi.org/10.1093/llc/fqs002

P. Gakis, C. Panagiotakopoulos, K. Sgarbas, C. Tsalidis V. Verykios. (2016). Design and construction of the Greek grammar checker. Digital Scholarship in the Humanities. 32 (2). 10.1093/llc/fqw025.

A. Ntoulas, S. Stamou, I. Tsakou, M. Tzagarakis, Ch. Tsalidis, A. Vagelatos: *Use of a Morphosyntactic Lexicon as the Basis for the Development of the Greek WordNet*, 2nd International Conference on Natural Language Processing, June 2000, Patras, Greece. Lecture Notes in Computer Science Springer Berlin / Heidelberg, Volume 1835/2000, 2000, pp 49-56

C. Tsalidis, A. Vagelatos, G. Orphanos: *Implementation of a Greek morphological lexicon for the biomedical domain*, International Conference on Information Communication Technologies in Health (ICICTH 2003), Samos, Greece, July 2003.

C. Tsalidis, A. Vagelatos, G. Orphanos: *An electronic dictionary as a basis for NLP tools: The Greek case*,11th Conference on Natural Language Processing (TALN '04), April 19 - 22, 2004, Fez, Morocco.

A. Vagelatos, E. Mantzari, G. Orphanos, Ch. Tsalidis, Ch. Kalamara, Ch. Diolis: *Implementing the NLP Infrastructure for Greek Biomedical Data Mining*, International Workshop A COMMON NATURAL LANGUAGE PROCESSING PARADIGM FOR BALKAN LANGUAGES, RANLP 2007 Conference, Borovets, Bulgaria, September 2007.

A. Vagelatos, T. Triantopoulou, C. Tsalidis, and D. Christodoulakis. 1995. Utilization of a lexicon for spelling correction in Modern Greek. In Proceedings of the 1995 ACM symposium on Applied computing (SAC '95). Association for Computing Machinery, New York, NY, USA, 267–271. DOI:https://doi.org/10.1145/315891.315979

Α. Ιορδανίδου & Μ. Πανταζάρα. Χτίζω Λέξεις. Εκδόσεις «Κοντύλι», 2010.