# ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

## ΣΧΟΛΗ ΨΗΦΙΑΚΗΣ ΤΕΧΝΟΛΟΓΙΑΣ
## ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

*ΕΞΟΡΥΞΗ ΓΝΩΣΗΣ ΑΠΟ ΕΛΛΗΝΙΚΑ ΚΕΙΜΕΝΑ ΚΑΙ ΣΥΓΚΡΙΣΗ ΑΛΓΟΡΙΘΜΩΝ ΕΞΑΓΩΓΗΣ ΛΕΞΕΩΝ-ΚΛΕΙΔΙΩΝ*

*Πτυχιακή εργασία*

Κλωνής-Σκέντζος Χαράλαμπος


Αθήνα, 2020

# ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ


**Τριμελής Εξεταστική Επιτροπή**

Επιβλέπων

Βαρλάμης Ηρακλής

Αναπληρωτής Καθηγητής, Τμήμα Πληροφορικής και Τηλεματικής, Χαροκόπειο Πανεπιστήμιο

Μέλη

Δημητρακόπουλος Γεώργιος

Επίκουρος Καθηγητής, Τμήμα Πληροφορικής και Τηλεματικής, Χαροκόπειο Πανεπιστήμιο

Δαλάκας Βασίλειος

Ε.ΔΙ.Π, Τμήμα Πληροφορικής και Τηλεματικής, Χαροκόπειο Πανεπιστήμιο

Ο Κλωνής-Σκέντζος Χαράλαμπος δηλώνω υπεύθυνα ότι:

1. Είμαι ο κάτοχος των πνευματικών δικαιωμάτων της πρωτότυπης αυτής εργασίας και από όσο γνωρίζω η εργασία μου δε συκοφαντεί πρόσωπα, ούτε προσβάλει τα πνευματικά δικαιώματα τρίτων.
2. Αποδέχομαι ότι η ΒΚΠ μπορεί, χωρίς να αλλάξει το περιεχόμενο της εργασίας μου, να τη διαθέσει σε ηλεκτρονική μορφή μέσα από τη ψηφιακή Βιβλιοθήκη της, να την αντιγράψει σε οποιοδήποτε μέσο ή/και σε οποιοδήποτε μορφότυπο καθώς και να κρατά περισσότερα από ένα αντίγραφα για λόγους συντήρησης και ασφάλειας.

# Ευχαριστίες

Με την ολοκλήρωση της πτυχιακής μου εργασίας θα ήθελα να ευχαριστήσω όλους όσους με βοήθησαν και με στήριξαν σε όλη τη διάρκεια των σπουδών μου αλλά και της εκπόνησης της εργασίας.

Ευχαριστώ θερμά τον επιβλέπων καθηγητή της πτυχιακής μου εργασίας, κύριο Βαρλάμη Ηρακλή, για την υποστήριξη, τις πολύτιμες συμβουλές του.

Θα ήθελα, επίσης, να ευχαριστήσω τα μέλη της εξεταστικής επιτροπής κ. Δημητρακόπουλο Γεώργιο και κ. Δαλάκα Βασίλειο που μου έκαναν την τιμή να αξιολογήσουν την προσπάθεια μου.

Τέλος, ένα μεγάλο ευχαριστώ στην Κάργατζη Ηλιάνα για την υποστήριξη και τις συμβουλές της όσον αφορά την ελληνική γραμματική αλλά και για τον χρόνο που αφιέρωσε για την επαλήθευση του αποτελέσματος της εργασίας.

# Περιεχόμενα

# Περίληψη

Αυτή η πτυχιακή εργασία κλήθηκε να συγκρίνει αλγορίθμους unsupervised εξαγωγής λέξεων κλειδιών και να μελετήσει τη συμπεριφορά τους σε κείμενα στην ελληνική γλώσσα. Στα πλαίσια της ανάγκης για προεπεξεργασία, χρησιμοποιήθηκε το spaCy και πάνω σε αυτό χτίστηκε ένας rule-based lemmatizer που αναζητά το λήμμα της λέξης με βάση τη γραμματική ανάλυση της. Με την υλοποίησή του, έγινε ο πρώτος lemmatizer στο spaCy που ακολουθεί αυτή την εμβαθυμένη λογική. Μετά την προεπεξεργασία, χρησιμοποιήθηκαν τα χαρακτηριστικά κειμένου Bag of Words και TF-IDF, και οι αλγόριθμοι TextRank, Yake και Rake. Συνδιαστικά, μέσω pointing system καταλήγουν σε μια μοναδική λίστα λέξεων κλειδιών που περιλαμβάνει μονογράμματα και διγράμματα. Η εργασία βρίσκει εφαρμογή στην ανάλυση κειμένων του διαδικτύου και πιο συγκεκριμένα ειδησεογραφικά κείμενα προσφέροντας στοχευμένη ανάκτηση πληροφορίας.

**Λέξεις κλειδιά**: [Εξόρυξη Γνώσης από Κείμενα, Εξαγωγή Φράσεων-Κλειδιών, Προεπεξεργασία Κειμένου, Ανάκτηση Πληροφορίας]

# Abstract

This thesis project compares unsupervised keyword extraction algorithms and studies their behavior on Greek texts. Due to the need of preprocessing, spaCy was used and based on it, a rule-based lemmatizer, who looks for a word's lemma based on it's grammar analysis, was built. With this implementation, it was the first lemmatizer in spaCy who follows such a deep strategy. After preprocessing, Bag of Words and TF-IDF features were used together with TextRank, Yake and Rake algorithms. Using a pointing system between their results, a single list of keywords is exported, containing unigrams and bigrams. This project is useful in analysing texts found on the web and more specifically news articles, offering targeted information retrieval.

**Keywords**: [Text Mining, Keyword Extraction, Text Preprocessing, Information Retrieval]

# Κατάλογος σχημάτων

# Κατάλογος πινάκων

# Συντομογραφίες

| | |
|---|---|
| BoW | Bag of Words |
| NLP | Natural Language Processing |
| TF-IDF | Term frequancy - Inverse document frequency |

# Γλωσσάρι

| | |
|---|---|
| Κανονικοποίηση | Η διαδικασία μετατροπής των δεδομένων σε μια κοινή βασική μορφή. |
| Corpus | Ένα αντιπροσωπευτικό σύνολο κειμένων το οποίο χρησιμοποιείται ως δείγμα. |
| Lemmatizer | Το εργαλείο εύρεσης της αρχικής μορφής μιας λέξης. |
| POS Tagger | Εργαλείο για την εύρεση του μέρους του λόγου μιας λέξης. |
| Unsupervised | Μέθοδος όπου ο αλγόριθμος δεν έχει εκπαιδευτεί με παράδειγμα αποτελεσμάτων. |

# 1 Εισαγωγή

## 1.1 Το πρόβλημα

Ποτέ άλλοτε η παρουσία της πληροφορίας στον παγκόσμιο ιστό δεν ήταν τόσο κυρίαρχη όσο σήμερα, και ποτέ άλλοτε δεν ήταν τόσο επιτακτική η ανάγκη για την πιο αποτελεσματική και κατάλληλη αξιοποίησή της. Ένα από τα σημαντικότερα προβλήματα στον τομέα της τεχνολογίας, είναι η διαχείριση του τέραστιου όγκου πληροφοριών που καθημερινά γεννάται, αποθηκεύεται και αναλύεται από εργαλεία, εανώ την ίδια στιγμή οι χρήστες των δεδομένων επιζητούν από αυτά όλο και πιο εξειδικευμένες πληροφορίες. Εξαιρώντας τα πολυμέσα, η πιο χρήσιμη πηγή πληροφοριών είναι τα κείμενα.

Συγκεκριμένα, στον χώρο της ειδησεογραφίας, λόγω του μεγάλου όγκου πληροφοριών και της αδιάληπτης παραγωγής νέας πληροφορίας, είναι αναγκαία η αυτοματοποίηση ορισμένων ενεργειών. Είναι συχνό φαινόμενο μηχανές αναζήτησης ειδήσεων να ανατρέχουν στις ήδη δημοσιευμένες ειδήσεις, να τις φιλτράρουν και να παρουσιάζουν στους χρήστες τους ειδήσεις υψηλής σημασίας ενώ ταυτόχρονα περιορίζουν τα fake news, ενημερώνοντάς τους έτσι, με πιο ακριβείς ειδήσεις και προσφέροντας τους έτσι μια καλύτερη εμπειρία χρήσης.

## 1.2 Συνεισφορά

Τέτοιου είδους αναλύσεις απαιτούν εξειδικευμένα εργαλεία με πρωταρχικό στόχο την απόσπαση των σημαντικών όρων είτε για λόγους περίληψης και σύνοψης είτε για λόγους κατηγοριοποίησης των κειμένων. Ένα τέτοιο εργαλείο καλείται να υλοποιήσει αυτή η πτυχιακή εργασία, εξειδικευμένο για την ελληνική γλώσσα, ικανό να προσφέρει τις ιδανικές λέξεις-κλειδιά για κάθε είδους άρθρα, ανεξάρτητα με τη θεματολογία τους.

Ο αρχικός στόχος αυτής της πτυχιακής εργασίας ήταν η μελέτη των μεθόδων εξαγωγής λέξεων-κλειδιών από κείμενα, και συγκεκριμένα, από άρθρα στην ελληνική γλώσσα. Αντικείμενο προς μελέτη αποτελεί ο τρόπος λειτουργίας τους και η σύγκριση της απόδοσής τους στα ελληνικά. Από την προεπεξεργασία του κειμένου (εξηγείται στο 2ο κεφάλαιο) προέκυψε πως τα εργαλεία εξόρυξης γνώσης από κείμενα συχνά δεν υποστηρίζουν την ελληνική γλώσσα, ενώ οι δυνατότητες όσων την υποστηρίζουν είναι περιορισμένες. Έτσι, προέκυψε η ανάγκη υλοποίησης ενός ολοκληρωμένου αλγορίθμου προεπεξεργασίας.

Επιλέχθηκε η χρήση της Python για την ανάπτυξή του, χρησιμοποιώντας το NLP εργαλείο spaCy. Στη συνέχεια, έχοντας πλέον πληροφορία σε κατάλληλη μορφή, ακολούθησε η έρευνα των μεθόδων εξαγωγής λέξεων-κλειδιών, η μελέτη του τρόπου λειτουργίας τους και η σύγκριση των αποτελεσμάτων τους. Έτσι, κατασκευάστηκε ένα ολοκληρωμένο εργαλείο, ικανό να διαβάσει απλά έγγραφα, να τα μελετήσει, να μάθει από αυτά πληροφορία και να τη χρησιμοποιήσει στην εξαγωγή των σημαντικών λέξεων.

## 1.3 Δομή της εργασίας

Η παρούσα πτυχιακή εργασία μπορεί να χωριστεί σε δύο μέρη. Το πρώτο μέρος αναφέρεται στα εργαλεία εξόρυξης γνώσης από κείμενα, ενώ το δεύτερο μέρος, στον αρχικό στόχο, τους αλγόριθμους εξαγωγής λέξεων-κλειδιών. Πιο αναλυτικά:

Στο 2ο Κεφάλαιο, παρουσιάζεται η θεωρία του text mining και γίνεται αναφορά των μεθόδων και των εργαλείων του. Παρουσιάζονται τα βήματα προεπεξεργασίας ενός κειμένου και η εξαγωγή χαρακτηριστικών κειμένου. Επιπλέον, παρουσιάζονται τα πιο διαδεδομένα εργαλεία, και αναφέρονται όσα υποστηρίζουν τη γλώσσα μας. Γίνεται μια σύντομη παρουσίαση του τομέα της εξόρυξης γνώσης από κείμενα καθώς και τα χαρακτηριστικά ενός κειμένου.

Στο 3ο Κεφάλαιο, μελετάται σε βάθος ο τρόπος λειτουργίας του lemmatizer των διαθέσιμων εργαλείων, το σκεπτικό του ελληνικού lemmatizer του spaCy καθώς και οι αλλαγές και προσθήκες στον κώδικα για την επίτευξη των βέλτιστων αποτελεσμάτων.

Στο 4ο Κεφάλαιο αναφέρονται οι μέθοδοι αυτοματοποιημένου και unsupervised keyword extraction, από απλές μετρικές, σε στατιστικές μεθόδους εώς και ολοκληρωμένους αυτόνομους αλγορίθμους, καθώς και ο τρόπος λειτουργίας τους με παραδείγματα.

Στο 5ο Κεφάλαιο συγκρίνονται τα αποτελέσματα των προαναφερόμενων αλγορίθμων, τα συμπεράσματα, καθώς και οι επόμενοι στόχοι αυτού του project.

# 2 Εξόρυξη γνώσης από κείμενα

## 2.1 Εισαγωγή

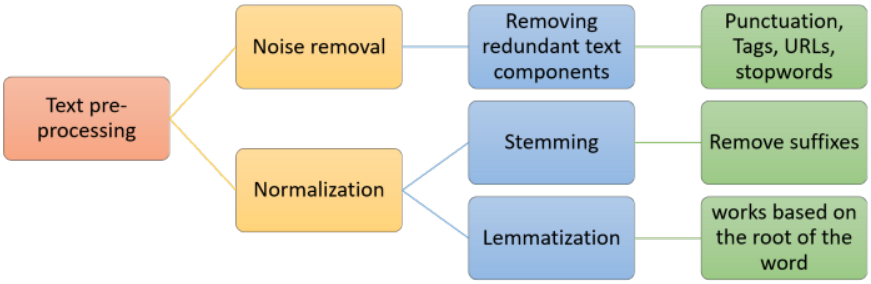
Με τον όρο εξόρυξη γνώσης από κείμενα, γνωστό στην επιστήμη της πληροφορικής ως Text Mining, εννοούμε τη διαδικασία άντλησης πληροφοριών μέσα από ένα κείμενο.

Κείμενα βρίσκονται παντού στο διαδίκτυο και η μορφή τους διαφέρει μιας και προέρχονται από πληθώρα πηγών (Koutropoulou, 2018): επαγγελματίες αρθρογράφοι, bloggers, ακόμη και οι απλοί χρήστες του διαδικτύου αφήνουν το στίγμα τους, με το πιο απλό παράδειγμα τα σχόλια στα κοινωνικά δίκτυα, ο όγκος των οποίων δυσκολεύει όλο και περισσότερο την εύρεση της προς αναζήτηση πληροφορίας. Οι αυξανόμενες ανάγκες για εξαγωγή πληροφορίας γίνονται ολοένα και πιο απαιτητικές στον κόσμο των επιχειρήσεων, όπου η πληροφορία πρέπει να είναι λεπτομερής και ακριβής, και η εξόρυξη γνώσης από δεδομένα (data mining) καλείται να τις καλύψει. Μέσω της ανάλυσης κειμένων, βγαίνουν στην επιφάνεια τα κομμάτια χρήσιμης πληροφορίας και με βάση αυτά, ικανοποιούνται οι απαιτήσεις ενός ευρέος φάσματος εφαρμογών, όπως το στοχευμένο marketing και τα συστήματα αντιμετώπισης καταστάσεων έκτακτης ανάγκης. Η εξόρυξη γνώσης από δεδομένα περιλαμβάνει πολλούς διαφορετικούς αλγορίθμους για να εκπληρωθούν διαφορετικές εργασίες. Μπορεί να μάθει από το κείμενο, να το περιλήψει, και να φέρει στην επιφάνεια την ωφέλιμη ουσία του.

Σήμερα, η εξόρυξη γνώσης από κείμενα είναι κάτι παραπάνω από ένα σύνολο εργαλείων τα οποία χρησιμοποιούνται για να ανακαλύψουν κρυμμένες πληροφορίες. Παρά την ύπαρξη πολλών εργαλείων, δεν υπάρχει ένα μοντέλο που να τα περιλαμβάνει όλα. Είναι άγνωστο αν στο μέλλον θα αναπτυχθεί κάποιος αλγόριθμος που θα είναι ικανός να επεξεργαστεί όλων των ειδών τα δεδομένα και να παρέχει όλες τις εφικτές πληροφορίες. Μέχρι τότε όμως, σκοπός των προγραμματιστών και των ερευνητών είναι η συνεχή ανάπτυξη νέων και βελτίωση ήδη υπαρχόντων εργαλείων στην προσπάθεια να αγγίξουν τον τέλειο αλγόριθμο.

## 2.2 Προεπεξεργασία

Η διαδικασία η οποία με είσοδο το αρχικό κείμενο παράγει ένα σύνολο σκέτων λέξεων μέσα στο οποίο βρίσκεται η πλεονότητα της πληροφορίας του κειμένου.




Σχήμα 1: Στάδια της προεπεξεργασίας

### 2.2.1 Κανονικοποίηση του κειμένου

Στο αρχικό στάδιο απομακρύνονται όσα σημεία στίξης και σύμβολα δεν προσφέρουν χρήσιμη πληροφορία, όπως πιθανές παρενθέσεις, εισαγωγικά κ.α. ενώ διατηρούνται για παράδειγμα οι παύλες, αφού συχνά δομούν μια σύνθετη έννοια. Επίσης, κάθε χαρακτήρας μετατρέπεται σε πεζός και αφαιρούνται τυχόν τόνοι, ώστε να μειωθούν οι περιπτώσεις όπου η ίδια λέξη παρουσιάζεται γραμμένη με διαφορετικό τρόπο, είτε λόγω κλίσης είτε λόγω της θέσης της στην πρόταση.

### 2.2.2 Απομάκρυνση Κοινών Λέξεων

Ο όρος "κοινές λέξεις" (stopwords) αναφέρεται στις λέξεις που εμφανίζονται με μεγάλη συχνότητα στα περισσότερα κείμενα. Είναι αυτές που, ενώ στον γραπτό και προφορικό λόγο είναι απαραίτητες για τη σύνδεση και τη δομή των προτάσεων, στην περίπτωση της ανάλυσης των κειμένων, η σημασιολογική τους αξία είναι σχεδόν μηδαμινή. Περιλαμβάνονται τα άρθρα, οι αντωνυμίες, τα επιρρήματα, οι προθέσεις, ακόμη και τα βασικά ρήματα. Με την αφαίρεση των λέξεων αυτών, μειώνεται ο όγκος των προς επεξεργασία δεδομένων, με αποτέλεσμα ο χρόνος που απαιτείται για την ολοκλήρωση της διαδικασίας να μειώνεται, ενώ ταυτόχρονα, οι λέξεις που διατηρούνται να είναι αυτές με μεγαλύτερη σημασιολογική αξία.

### 2.2.3 Επιλογή λέξεων με βάση το μέρος του λόγου

Ένας ακόμη τρόπος περιορισμού των λέξεων είναι το φιλτράρισμά τους μέσω του μέρους του λόγου. Για να βρεθεί αυτό, χρησιμοποιούνται NLP εργαλεία που ονομάζονται POS Tagger. Ανάλογα με το εκάστοτε ζητούμενο, μπορούν να επιλεχθούν διαφορετικά μέρη του λόγου ως αποδεκτά και οι λέξεις να περιοριστούν σε αυτά.

### 2.2.4 Οριοθέτηση μήκους λέξεων

Η πιο απλή αλλά και αποτελεσματική μέθοδος, αφορά το μήκος της κάθε λέξης. Αν αυτό είναι πολύ μικρό, είναι πιθανό η λέξη αυτή να μην βοηθάει την ανάλυση του κειμένου, κι ας μην υπάρχει στην προκαθορισμένη λίστα stopwords. Αντίθετα, αν είναι πολύ μεγάλο, ίσως οφείλεται σε τυπογραφικό λάθος ή να έχει άλλη έννοια (π.χ. υπερσύνδεσμος ιστοσελίδας).

### 2.2.5 Στελέχωση

Είναι η διαδικασία ομαδοποίησης των υποψήφιων προς εξαγωγή λέξεων με βάση τη ρίζα τους (Stemming). Με την αφαίρεση της κατάληξης της λέξης, παραμένει μόνο το θέμα της ανεξαρτήτως αριθμού, γένους, πτώσης κλπ. Με τον τρόπο αυτό, αν και περιορίζεται η επανάληψη λέξεων της ίδιας ρίζας, υπάρχει το ενδεχόμενο να αλλοιωθεί η αρχική σημασία κάποιων λέξεων. Για παράδειγμα, η λέξη "χαρακτηριστικών" θα επιστρέψει το θέμα της λέξης, "χαρακτη-".

### 2.2.6 Λημματοποίηση

Την λύση στο προαναφερόμενο ζήτημα δίνει η Λημματοποίηση (Lemmatization). Με στόχο την επαναφορά κάθε λέξης στην αρχική της μορφή, εφαρμόζονται απλές ή σύνθετες διαδικασίες, ανάλογα με τον τρόπο λειτουργίας τους, όπως εξηγούνται στο 3ο κεφάλαιο. Αντίστοιχο παράδειγμα, η λέξη "χαρακτηριστικών" θα πρέπει να φέρει το αποτέλεσμα "χαρακτηριστικά".

## 2.3 Features

Μπορεί το κείμενο να είναι μια πολύ εύκολη προς ανάγνωση πληροφορία για τον άνθρωπο, για τις μηχανές όμως είναι περίπλοκο να το κατανοήσουν, καθώς είναι αδόμητο. Για να φανεί χρήσιμη η πληροφορία αυτή, χρειάζεται το κείμενο να μετατραπεί σε κάποια κατανοητή μορφή για τους υπολογιστές, όπως πίνακες ή διανύσματα από χαρακτηριστικά (features). Ακολουθούν οι πιο γνωστές τεχνικές εξαγωγής χαρακτηριστικών από κείμενα.

### 2.3.1 Bag of Words

Η πιο απλή μέθοδος εξαγωγής χαρακτηριστικών είναι η BoW (Bag of Words), καθώς βγάζει το χαρακτηριστικό της παρουσίας λέξεων. Ονομάζεται έτσι μιας και δεν παίζει ρόλο η συχνότητα της λέξης ή οι σειρά τους, παρά μόνο η ύπαρξή τους στην ίδια ομάδα λέξεων. Αποτελεί την παρουσίαση των κειμενικών δεδομένων και περιλαμβάνει το λεξικό των γνωστών λέξεων και τη συχνότητα της κάθε λέξης. Θα χρησιμοποιηθεί το ακόλουθο άρθρο ως παράδειγμα:

"Από τη Διεύθυνση Μεταφορών και Επικοινωνιών Δυτικής Θεσσαλονίκης της Περιφέρειας Κεντρικής Μακεδονίας ανακοινώνεται ότι οι εξετάσεις απόκτησης πιστοποιητικού επαγγελματικής επάρκειας οδικών μεταφορέων θα διεξαχθούν με πρώτη ημέρα εξέτασης τη Δευτέρα 29 Ιουνίου 2020. Οι εξετάσεις θα πραγματοποιηθούν στην αίθουσα ειδικών εξετάσεων ADR και στην αίθουσα εξέτασης Κ.Ο.Κ. της Διεύθυνσης Μεταφορών και Επικοινωνιών Δυτικής Θεσσαλονίκης (Καλοχώρι). Λόγω των μέτρων του κορωνοιού, ο ακριβής αριθμός των ημερών που θα διαρκέσουν οι εξετάσεις, καθώς και οι ώρες προσέλευσης θα καθοριστούν, αφού προηγουμένως οριστικοποιηθεί ο ακριβής αριθμός των υποψηφίων."

'διευθυνση': 0, 'μεταφορες': 1, 'επικοινωνιες': 2, 'δυτικη': 3, 'θεσσαλονικη': 4, 'περιφερεια': 5, 'κεντρικη': 6, 'μακεδονια': 7, 'εξετασεις': 8, 'αποκτηση': 9, 'πιστοποιητικο': 10, 'επαγγελματικη': 11, 'επαρκεια': 12, 'οδικες': 13, 'μεταφορεις': 14, 'ημερα': 15, 'εξεταση': 16, 'δευτερα': 17, 'ιουνιος': 18, '2020': 19, 'αιθουσα': 20, 'ειδικες': 21, 'Καλοχώρι': 22, 'μετρα': 23, 'κορωνοιος': 24, 'αριθμος': 25, 'ωρες': 26, 'προσελευση': 27, 'υποψηφια': 28

Στην BoW μορφή κάθε ζευγάρι αριθμών αποτελεί μια λέξη. Το πρώτο ψηφίο αντιστοιχεί σε μια από τις παραπάνω αριθμημένες λέξεις, ενώ το δεύτερο στο πόσες φορές εμφανίζεται η λέξη αυτή.

```
[(0, 2), (1, 2), (2, 2), (3, 2), (4, 2), (5, 1), (6, 1), (7, 1), (8, 4),
(9, 1), (10, 1), (11, 1), (12, 1), (13, 1), (14, 1), (15, 2), (16, 2),
(17, 1), (18, 1), (19, 1), (20, 2), (21, 1), (22, 1), (23, 1), (24, 1),
(25, 2), (26, 1), (27, 1), (28, 1)]
```

### 2.3.2 TF-IDF

Το πρόβλημα της προαναφερόμενης μεθόδου, BoW, είναι πως οι πιο συχνές λέξεις είναι αυτές που κυριαρχούν στα δεδομένα. Η συχνότητα εμφάνισης μιας λέξης δεν συνδέεται απόλυτα με την σημασιολογική αξία της. Μια πιο ολοκληρωμένη προσέγγιση προσφέρει το TF-IDF (Term Frequency - Inverse Document Frequency), ένας τρόπος αξιολόγησης λέξεων που χρησιμοποιείται κυρίως στην ανάκτηση πληροφοριών και στην περίληψη. Δείχνει την σχετικότητα ενός όρου μέσα στο δωσμένο κείμενο.

Η λογική του είναι πως όσο περισσότερες φορές εμφανίζεται μια λέξη στο κείμενο, τόσο περισσότερο μεγαλώνει η αξία της (TF). Παράλληλα όμως, αν η λέξη αυτή εμφανίζεται συχνά και σε υπόλοιπα κείμενα του corpus, θα σημαίνει πως πρόκειται για κάποια κοινή λέξη, όχι τόσο σχετική ή ουσιώδης για το αρχικό κείμενο (IDF). Αυτό σημαίνει πως όσο μεγαλύτερη είναι η συλλογή κειμένων, τόσο πιο ακριβή θα είναι τα αποτελέσματα του TF-IDF. Κάθε λέξη ή όρος, αποκτά το προσωπικό του TF και IDF σκορ. Το παράγωγό τους, ονομάζεται TF-IDF σκορ του όρου. Όσο μεγαλύτερο το TF-IDF, τόσο πιο χρήσιμη είναι μία λέξη, και το αντίστροφο.

Αν ταξινομήσει κανείς το TF-IDF όλων των λέξεων ενός κειμένου κατά φθίνουσα σειρά, θα αποκτήσει τις μοναδικότερες λέξεις του κειμένου, κάτι που σε πολλές περιπτώσεις αποτελεί τη σημασία των λέξεων-κλειδιών. Γι' αυτό, στα πλαίσια της εργασίας το TF-IDF θα χρησιμοποιηθεί και ως αλγόριθμος εξαγωγής λέξεων κλειδιών. Ακολουθεί αναλυτικό παράδειγμα στο κεφάλαιο 4.2.2.

## 2.4 Εργαλεία για text mining

Μιας και το NLP γίνεται όλο και πιο διαδεδομένο στον χώρο της πληροφορικής, τα εργαλεία και οι δυνατότητές τους βελτιώνονται και πληθαίνουν. Τα εργαλεία αυτά προσφέρουν με τις λειτουργίες τους στους προγραμματιστές μια πιο εύκολη εμπειρία χρήσης. Παλαιότερα, ήταν απαραίτητη η προηγμένη φιλολογική γνώση για να μπορέσει κανείς να ασχοληθεί με την επεξεργασία φυσικής γλώσσας.

Ακολουθούν τα πιο δημοφιλή NLP εργαλεία που χρησιμοποιούνται για text mining με χρήση της Python:

**Natural Language Toolkit - NLTK**

Το NLTK είναι το παλαιότερο και πιο γνωστό παγκοσμίως μοντέρνο εργαλείο NLP. Πρωτοεμφανίστηκε το 2001 στο University of Pennsylvania (Bird Steven & Klein, 2001), είναι ανοιχτού κώδικα και υποστηρίζει όλες τις απαραίτητες λειτουργίες μέσω πολλών αλγορίθμων. Χρησιμοποιείται από πανεπιστήμια σε όλο τον κόσμο για εκμάθηση του NLP μιας και απαιτεί από τον προγραμματιστή να το μελετήσει βαθύτερα και να συμβουλεύεται συχνά το documentation για να φτάσει στο επιθυμητό του αποτέλεσμα. Σημαντική λεπτομέρια είναι πως η πληροφορία που υπάρχει και επεξεργάζεται είναι τύπου string, κάτι που δεν συνηθίζεται σε αντικειμενοστρεφείς γλώσσες όπως η Python. Έχει αναβαθμιστεί πολλές φορές μέσα σε αυτά τα χρόνια αλλά δεν έχει φτάσει τους ανταγωνιστές του σε ταχύτητα και απλότητα.

Οι υποστηριζόμενες γλώσσες διαφέρουν ανάλογα με τη λειτουργία, καθώς συχνά χρησιμοποιείται άλλος αλγόριθμος για την κάθε γλώσσα μέσω επεκτάσεων (extensions) (Padmanabhan, 2019). Κάποιες λειτουργίες όπως ο χωρισμός του κειμένου σε προτάσεις υποστηρίζουν τις περισσότερες γλώσσες καθώς δεν μελετούν γραμματικά ή σημασιολογικά τη γλώσσα, παρά μόνο κοινά σύμβολα. Η γλώσσα που υποστηρίζεται σε όλες τις λειτουργίες χωρίς την προσθήκη εξωτερικών επεκτάσεων είναι η αγγλική.

**spaCy**

Το spaCy είναι μια νέα βιβλιοθήκη ανοιχτού κώδικα και ο βασικός ανταγωνιστής του NLTK, οπού το ξεπερνά σε ταχύτητα στις περισσότερες περιπτώσεις. Αυτό οφείλεται στο γεγονός ότι μέρη της βιβλιοθήκης είναι γραμμένα σε Cython, εξού και το όνομα της βιβλιοθήκης. Δημοσιεύθηκε από τους Matthew Honnibal και Ines Montani (Honnibal, 2015). Απευθύνεται κυρίως στην ανάπτυξη λογισμικού για περιβάλλοντα παραγωγής. Αντίθετα από το NLTK, κάθε λειτουργία αναπαρίσταται με τα αντίστοιχα αντικείμενα (π.χ. αντικείμενο doc για το κάθε document). Είναι πιο εύκολο στην εκμάθηση και χρήση καθώς διαθέτει ένα μοναδικό και βελτιστοποιημένο εργαλείο για κάθε λειτουργία.

Πλέον υποστηρίζει 58 γλώσσες με έτοιμα εκπαιδευμένα μοντέλα για 15 από αυτές, συμπεριλαμβανομένης και της ελληνικής. Επιπλέον διαθέτει μοντέλο για ταυτόχρονη επεξεργασία πολλαπλών γλωσσών (multi-language).

**scikit-learn**

Παρέχει στους προγραμματιστές ένα ευρύ φάσμα αλγορίθμων για την κατασκευή μοντέλων μηχανικής μάθησης (machine learning). Χρησιμοποιεί τα χαρακτηριστικά των κειμένων που αναφέρθηκαν στο προηγούμενο υποκεφάλαιο, όπως το BoW και επιλέγεται συχνά για την κατηγοριοποίηση (classification) κειμένων. Καθώς όμως δεν χρησιμοποιεί νευρωνικά δίκυα, είναι καλύτερο να γίνεται χρήση των υπόλοιπων εργαλείων για την προετοιμασία του κειμένου και να χρησιμοποιείται στο τέλος για την κατασκευή των μοντέλων. (Kozaczko, 2018)

**gensim**

Είναι μια βιβλιοθήκη με ειδικότητα στην αναγνώριση σημασιολογικής ομοιότητας μεταξύ δυο κειμένων μέσω μοντελοποίησης διανυσματικού χωρου (vector space modeling). Έχει την ικανότητα να διαχειριστεί μεγάλο όγκο κειμένων με εντυπωσιακή διαχείρηση μνήμης και γρήγορους χρόνους εκτέλεσης λόγω της χρήσης του NumPy. Αντίστοιχα με το scikit-learn, δεν διαθέτει αρκετά εργαλεία για να καλύψει όλες τις ανάγκες της εξόρυξης γνώσης από κείμενα και έτσι η χρήση περισσότερων βιβλιοθηκών είναι αναγκαία. Περισσότερες Πληροφορίες

**Polyglot**

Μια βιβλιοθήκη με σκοπό την υποστήριξη των περισσότερων γλωσσών, με τις βασικές λειτουργίες να υποστηρίζουν κοντά στις 200 γλώσσες. Σε πιο δύσκολες λειτουργίες όπως το POS-Tagging όμως, υποστηρίζει μόνο 16, και τα ελληνικά δεν είναι μια από αυτές. Η εμπειρία χρήσης του θυμίζει την απλότητα του spaCy, η ταχύτητά του ανταγωνίζεται τις υπόλοιπες βιβλιοθήκες χάρη στη χρήση του NumPy, αλλά προτιμάται για κείμενα σε γλώσσες που δεν υποστηρίζονται από άλλα εργαλεία.

**Classical Language Toolkit - CLTK**

Έχει ονομαστεί εμπνευσμένο από το NLTK, αλλά απευθύνεται σε διαφορετικό κοινό μιας και εστιάζει στην μελέτη και επεξεργασία των γλωσσών της αρχαίας, κλασσικής και μεσαιωνικής Ευρασίας. Χρησιμοποιεί επιλεγμένους αλγορίθμους του NLTK. Αν και δεν είναι από τις πιο γνωστές βιβλιοθήκες, γίνεται αναφορά λόγω της υποστήριξης των ελληνικών. Αν και οι περισσότερες λειτουργίες αφορούν την Αρχαία Ελληνική, οι βασικές λειτουργίες φέρνουν σωστά αποτελέσματα και στη Νέα Ελληνική.

Ολοκληρωμένη υποστήριξη παρέχει για τις γλώσσες Αρχαία Ελληνικά, Λατινικά, Αρχαία Ακκαδικά (Μεσσοποταμία) και τις Γερμανικές γλώσσες (Germanic languages). Αξίζει να αναφερθεί η (μερική) υποστήριξη γλωσσών όπως τα Αραμαϊκά, τα Αρχαία Αιγυπτιακά, τα Αγγλοσαξονικά, τα Περσικά και τα Κλασικά Θιβετιανά. (Johnson, Burns, Stewart, & Cook, 2014--2020)

# 3 Ανάλυση και βελτιστοποίηση εργαλείων εξόρυξης γνώσης

## 3.1 spaCy και ελληνικά

Για την προεπεξεργασία του κειμένου σε αυτή τη πτυχιακή εργασία επιλέχθηκε η βιβλιοθήκη spaCy. Σε αυτή, η υποστήριξη της ελληνικής γλώσσας υλοποιήθηκε στα πλαίσια ενός πρόσφατου GSOC (Google Summer Of Code) Project και συγκεκριμένα το 2018, υπό την αιγίδα της ΕΕΛΛΑΚ (Οργανισμός Ανοιχτών Τεχνολογιών στην Ελλάδα). Ο τότε φοιτητής, προγραμματιστής του project και συγγραφέας του documentation είναι ο Γιάννης Δάρας.

Στα πλαίσια του project αυτού, υλοποιήθηκαν 13 λειτουργίες, οι οποίες περιλαμβάνονται στα τρια μοντέλα της ελληνικής γλώσσας που παρήχθηκαν. Η διαφορά του small από τα medium και large μοντέλα είναι η μη υποστήριξη ανίχνευση ομοιότητας μεταξύ κειμένων, όπου γίνεται κάνοντας χρήση word vectors.

**Tokenizer - Χωρισμός λέξεων**

Χρησιμοποιείται για να σπάει τις προτάσεις σε λέξεις (tokens).

Είσοδος: Θέλω να μου σπάσεις αυτήν την πρόταση σε κομμάτια
Έξοδος: [Θέλω, να, μου, σπάσεις, αυτήν, την, πρόταση, σε, κομμάτια]

**Lemmatizer - Λημματοποιητής**

Διατηρεί το λήμμα της κάθε λέξης του κειμένου, δηλαδή την μετατρέπει στην αρχική της μορφή.

Είσοδος: Τα σύμβολα του αγώνα.
Έξοδος: Original token: Τα , Lemma: τα
Original token: σύμβολα , Lemma: σύμβολο
Original token: του , Lemma: του
Original token: αγώνα , Lemma: αγώνας
Original token: . , Lemma: .

**Sentence Splitter - Διαχωριστής προτάσεων**

Χρησιμοποιείται για να σπάει το κείμενο σε προτάσεις.

Είσοδος: Αυτή είναι μια πρόταση. Αυτή είναι μια δεύτερη πρόταση.
Και αυτή μια τρίτη πρόταση.
Έξοδος: [ Αυτή είναι μια πρόταση., Αυτή είναι μια δεύτερη πρόταση.,
Και αυτή μια τρίτη πρόταση.]

### Λίστα stop words

Η λίστα δημιουργήθηκε με βάση την Ελληνική Βικιπαίδια και τις 300 πιο συχνές της λέξεις και εμπλουτίστηκε πραγματοποιώντας αντίστοιχη διαδικασία για τους ελληνικούς υπότιτλους στο Open Subtitles. Τέλος, με πηγή την λίστα stop words του Dr. Holger Bagola (Bagola, 2002) κατέληξε μια λίστα μήκους 663 λέξεων. Η λίστα βρίσκεται εδώ.

### Norm exceptions - Εξαιρέσεις

Η λίστα αυτή χρησιμεύει στην κανονικοποίηση του κειμένου έτσι ώστε λέξεις γραμμένες με διαφορετικό τρόπο να αντιμετωπίζονται ως ίδιες. Η λίστα αυτή παράχθηκε αυτόματα με χρήση λεξικού. Παράδειγμα λέξης γραμμένης διαφορετικά αποτελεί το "αιρκοντίσιον": "ερκοντίσιον", ενώ παράδειγμα ορθογραφικού λάθους που με αυτόν τον τρόπο διορθώνεται αποτελεί το "αλλοιώς": "αλλιώς". Η λίστα βρίσκεται εδώ.

### Named Entities Recognition - Αναγνώριση Οντοτήτων

Το NER αποτελεί το εργαλείο που χαρακτηρίζει τις λέξεις ανάλογα με την οντότητα στην οποία ανήκουν. Τα πιθανά αποτελέσματα είναι:

ORG: Εταιρίες, ινστιτούτα, πολιτικά κόμματα κλπ.
PRODUCT: Αντικείμενα, οχήματα, φαγητά κλπ.
PERSON: Άνθρωποι, συμπεριλαμβανομένου των φανταστικών χαρακτήρων.
EVENT: Μάχες, πόλεμοι, αθλητικοί αγώνες, φυσικά φαινόμενα κλπ.
GPE: Γεωπολιτικές Οντότητες, κράτη, πόλεις κλπ.
LOC: Τοποθεσίες που δεν ανήκουν στην προηγούμενη κατηγορία, όπως οροσειρές, ποτάμια κλπ.

Καθώς δεν υπήρχε άλλο dataset με ελληνικές οντότητες, κατασκευάστηκε ένα με χρήση του Prodigy.

### Lexical attributes - Λεξικολογικά χαρακτηριστικά

Γίνεται έλεγχος σε κάθε κείμενο για το αν λέξεις αποτελούν ειδικά tokens οπως υπερσύνδεσμοι ή νούμερα.

Είσοδος: Η ιστοσελίδα για το demo μας είναι: https://nlp.wordames.gr
Έξοδος: Url: https://nlp.wordames.gr

**POS Tagger**

Όπως έχει ήδη εξηγηθεί, το εργαλείο αυτό επιστρέφει το μέρος του λόγου της κάθε λέξης.

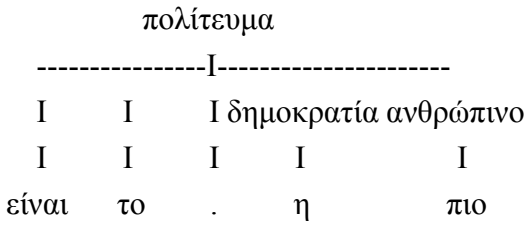
Είσοδος: Η δημοκρατία είναι το πιο ανθρώπινο πολίτευμα.
Έξοδος: Token: Η Tag: DET
Token: δημοκρατία Tag: NOUN
Token: είναι Tag: AUX
Token: το Tag: DET
Token: πιο Tag: ADV
Token: ανθρώπινο Tag: ADJ
Token: πολίτευμα Tag: NOUN
Token: . Tag: PUNCT

Η λίστα με όλες τις συντομογραφίες για τα μέρη του λόγου και τις εξηγήσεις τους μπορεί να βρεθεί εδώ.

**DEP Tagger - Συντακτική εξάρτηση**

Με το εργαλείο αυτό μπορεί να γίνει είτε ανάλυση είτε οπτικοποίηση μιας πρότασης με βάση τις συντακτικές εξαρτήσεις των λέξεων.

Είσοδος: η δημοκρατία είναι το πιο ανθρώπινο πολίτευμα.
Έξοδος:




**Noun chunks - Βασικές προτάσεις ουσιαστικών**

Ένας ακόμη τρόπος για σπάσιμο του κειμένου σε φράσεις που περιλαμβάνει ένα ουσιαστικό και όσες λέξεις το περιγράφουν. Για παράδειγμα, "Η μεγαλύτερη εταιρία πληροφορικής παγκοσμίως".

**Sentiment analyzer - Αναλυτής συναισθημάτων**

Αναλύει το κείμενο και επιστρέφει το ποσοστό υποκειμενικότητας, το κύριο συναίσθημα και το ποσοστό του. Υποστηρίζει τα συναισθήματα θυμού, αηδίας, φόβου, χαράς, λύπης και έκπληξης. Αξίζει να σημειωθεί πως όταν γράφτηκε δεν υπήρχε άλλος αναλυτής συναισθημάτων μέσα σε εκπαιδευμένο μοντέλο άλλης γλώσσας στο spaCy.

Είσοδος: Έχω μείνει έκπληκτος! Πώς γίνεται αυτό; Η έκπληξη είναι τόσο μεγάλη!
Α, τώρα εξηγούνται όλα.
Έξοδος: Subjectivity: 16.66%, Main emotion: surprise, Emotion score: 33.33%

**Topic Classifier - Κατηγοριοποιητής Θέματος**

Χαρακτηρίζει την κατηγορία στην οποία ανήκει το κείμενο. Υποστηρίζει τις κατηγορίες Αθλητικών, Επιστήμης, Παγκόσμιων Ειδήσεων, Ελληνικών Ειδήσεων, Πολιτικής, Τέχνης και Υγείας. Παρομοίως, τη στιγμή που γράφτηκε το εργαλείο δεν υπήρχε αντίστοιχο για άλλη γλώσσα.

Τα παραδείγματα καθώς και περισσότερες πληροφορίες μπορούν να βρεθούν στο documentation του gsoc2018-spacy project.

## 3.2 Λημματοποιητές στο spaCy

Οι περισσότερες γλώσσες στο spaCy χρησιμοποιούν lookups για την λημματοποίηση. Αυτό σημαίνει πως για κάθε μοντέλο των γλωσσών αυτών, υπάρχει ένα τεράστιο κειμενικό αρχείο που περιέχει τις αντιστοιχίες όλων των λέξεων με το λήμμα τους, την αρχική τους μορφή.

Κατά τον συγγραφέα και προγραμματιστή, αυτή η προσέγγιση δεν είναι η κατάλληλη για την ελληνική γλώσσα. Συγκεκριμένα, στο documentation αναφέρει τις γραμματικές πολυπλοκότητες των ελληνικών που καθιστούν την παραγωγή ενός τεράστιου lookup αρχείου ώστε να καλύπτει όλους τους χρόνους, τις κλίσεις και τα πρόσωπα των ρημάτων, ουσιαστικών και επιθέτων.

| |
|---|
| ”aba”: ”abar”, |
| ”ababa”: ”abar”, |
| ”ababais”: ”abar”, |
| ”ababan”: ”abar”, |
| ”ababanes”: ”ababán”, |
| ”ababas”: ”abar”, |
| ”ababoles”: ”ababol”, |
| ”ababábites”: ”ababábite” |

Πίνακας 1: Απόσπασμα της λίστας του ισπανικού λημματοποιητή

Αντίθετα από αυτόν τον τρόπο, για την ελληνική γλώσσα επιλέχθηκε μια προσέγγιση βασισμένη σε κανόνες. Αυτό υλοποιείται με λίστες κανόνων ξεχωριστές για κάθε μέρος του λόγου, και βασίζεται στην αντικατάσταση της κατάληξης. Για παράδειγμα αν η λέξη είναι "(της) πέτρας", αναγνωρίζεται από τον POS Tagger ως ουσιαστικό και ψάχνοντας τον κατάλληλο κανόνα, θα αντικαταστήσει την κατάληξη -ας, με την κατάληξη -α.

Η μέθοδος των lookups ναι μεν προσφέρει απόλυτη ακρίβεια στην εύρεση λήμματος, αλλά καθιστά τη διαδικασία πιο αργή, απαιτεί μεγάλο όγκο αντιστοιχιών και δεν μπορεί να επεξεργαστεί λέξεις που δεν υπάρχουν ήδη καταγεγραμμένες στη λίστα. Από την άλλη, η μέθοδος βασισμένη σε κανόνες μπορεί να μην είναι τόσο ακριβείς, αλλά προσφέρει ταχύτητα στην υλοποίηση και στην εκτέλεση και απαιτεί ελάχιστο χώρο για τους κανόνες. Μπορεί να επεξεργαστεί λέξεις που δεν έχει συναντήσει, αντιμετοπίζοντάς τις με τον ίδιο ακριβώς τρόπο, κρίνοντας από το μέρος του λόγου και την κατάληξή τους.

Σύμφωνα με τις τελευταίες ενημερώσεις του spaCy, ο ελληνικός lemmatizer διαφέρει από τον κοινό, είναι βελτιστοποιημένος για την ελληνική γλώσσα και πιο γρήγορος στην εξαγωγή των λημμάτων, καθώς ελέγχει το ενδεχόμενο η λέξη να είναι ήδη λήμμα. Επιπλέον, από την έκδοση 2.2 τα lookups περιλαμβάνουν μαζί με τις λίστες με την ένα προς ένα αντιστοιχία λέξεων (πλέον ονομαζόμενες lemma_lookups) και λίστες με κανόνες ως lemma_rules.

Ακολουθούν ενδεικτικά οι κανόνες των επιθέτων:

| ADJECTIVE_RULES = |
|---|
| ”οί”, ”ός”, ( καρδιακοί -> καρδιακός. Ονομαστική πλ. σε -ός. (m) ) |
| ”ών”, ”ός”, ( καρδιακών -> καρδιακός. Γενική πλ. σε -ός. (m) ) |
| ”ού”, ”ός”, ( καρδιακού -> καρδιακός. Γενική εν. σε -ός. (m) ) |
| ”ή”, ”ός”, ( καρδιακή -> καρδιακός. Ονομαστική εν. σε -ή. (f) ) |
| ”ής”, ”ός”, ( καρδιακής -> καρδιακός. Γενική εν. σε -ή. (f) ) |
| ”ές”, ”ός”, ( καρδιακές -> καρδιακός. Ονομαστική πλ. σε -ή. (f) ) |
| ”οι”, ”ος”, ( ωραίοι -> ωραίος. Ονομαστική πλ. σε -ος. (m) ) |
| ”ων”, ”ος”, ( ωραίων -> ωραίος. Γενική πλ. σε -ος. (m) ) |
| ”ου”, ”ος”, ( ωραίου -> ωραίος. Γενική εν. σε -ος. (m) ) |
| ”ο”, ”ος”, ( ωραίο -> ωραίος. Ονομαστική εν. σε -ο. (n) ) |
| ”α”, ”ος”, ( χυδαία -> χυδαίος. Ονομαστική πλ. σε -ο. (n) ) |
| ”ώδη”, ”ώδες”, ( δασώδη -> δασώδες. Ονομαστική πλ. σε -ώδες. (n) ) |
| ”ύτερη”, ”ός”, ( καλύτερη -> καλός. Συγκριτικός βαθμός σε -ή. (f) ) |
| ”ύτερης”, ”ός”, ( καλύτερης -> καλός. (f) ) |
| ”ύτερων”, ”ός”, ( καλύτερων -> καλός. (f) ) |
| ”ύτερος”, ”ός”, ( καλύτερος -> καλός. Συγκριτικός βαθμός σε -ός. (m) ) |
| ”ύτερου”, ”ός” ( καλύτερου -> καλός. (m) ) |

Πίνακας 2: Απόσπασμα των κανόνων του ελληνικού λημματοποιητή

Όλο το σύνολο κανόνων μπορεί να βρεθεί εδώ.

Επιπλέον, στο lookup της γλώσσας περιλαμβάνεται λίστα εξαιρέσεων (lemma_exc) και λίστα γνωστού λεξιλογίου (lemma_index), ανά μέρος του λόγου αντίστοιχα. Η λίστα των εξαιρέσεων χρησιμεύει στην απλοποίηση συνηθισμένων λέξεων (π.χ. "ελάχιστος":"λίγος"), και όπως είναι αναμενόμενο, στην λημματοποίηση λέξεων που αποτελούν εξαιρέσεις (π.χ. "πω":"λέω"). Το γνωστό λεξιλόγιο βοηθά στην ακρίβεια του lemmatizer, καθώς αν μια λέξη με την διορθωμένη της κατάληξη περιλαμβάνεται στο γνωστό λεξιλόγιο, η λημματοποίηση κρίνεται επιτυχής και δεν δοκιμάζονται άλλες καταλήξεις.

## 3.3 Λημματοποιητής της εργασίας

Αν και η υλοποίηση του ελληνικού lemmatizer είναι εξελιγμένη, το spaCy και οι δυνατότητές του επιτρέπουν την βελτιστοποίηση των αποτελεσμάτων της διαδικασίας.

Το πρόβλημα της ανακρίβειας του συγκεκριμένου rule-based lemmatizer πηγάζει από το γεγονός πως δεν λαμβάνει υπ'όψιν του τα γραμματικά χαρακτηριστικά των λέξεων. Στις λέξεις που περιλαμβάνονται στο γνωστό λεξιλόγιο αυτό δεν αποτελεί πρόβλημα. Όμως, λέξεις της καθομιλουμένης, ονόματα και συγκεκριμένες ορολογίες που συχνά συναντάμε να χρησιμοποιούνται σε κείμενα ανερτημένα στο διαδίκτυο, δεν περιλαμβάνονται στο λεξιλόγιο. Έτσι, ο αλγόριθμος δεν μπορεί να προβλέψει σωστά την κατάληξη της λέξεις που πρέπει να εφαρμόσει, και περαιτέρω μελέτη είναι αναγκαία. Πέραν από το μέρος του λόγου (POS), το spaCy παρέχει και μια αναλυτικότερη γραμματική περιγραφή που περιλαμβάνει στοιχεία όπως το γένος, τον αριθμό, την πτώση κλπ. Ονομάζεται TAG και τα ελληνικά μοντέλα το υποστηρίζουν.

Αυτό μας επιτρέπει την δημιουργία στοχευμένων κανόνων για την αντικατάσταση της κατάληξης. Για παράδειγμα, η λέξη "(του) ιουνίου" θα επιστρέψει το POS = 'NOUN', ενώ το TAG = 'NOUN_Case=Gen|Gender=Masc|Number=Sing', δηλαδή Ουσιαστικό, Γενικής Πτώσης, Αρσενικού Γένους και Ενικού Αριθμού.

Με βάση αυτή τη λογική γράφτηκαν νέοι προσαρμοσμένοι κανόνες, ενώ οι παλιοί διατηρήθηκαν σε περίπτωση που δεν βρεθεί κανόνας για τη συγκεκριμένη κατηγορία. Καθώς οι περιπτώσεις λέξεων στην κλιτική πτώση ήταν ελάχιστες, δεν γράφτηκαν κανόνες για αυτές. Αντίστοιχα, σε ορισμένες περιπτώσεις οι λέξεις στην αιτιατική διατηρούσαν την ίδια μορφή με την ονομαστική. Στις περιπτώσεις αυτές η λέξη θεωρείται λήμμα και επιστρέφεται στη μορφή που είναι. Οι νέοι κανόνες είναι γραμμένοι για ουσιαστικά, αλλά πάνω σε αυτούς υποστηρίζουν επίσης κύρια ουσιαστικά και επίθετα.

Όλοι οι κανόνες του λημματοποιητή της εργασίας βασίζονται στους γραμματικούς κανόνες όπως εξηγούνται στο σχολικό βιβλίο "Γραμματική Νέας Ελληνικής Γλώσσας" των Σωφρόνη Χατζησαββίδη και Αθανασία Χατζησαββίδου για τις Α-Β-Γ τάξεις του Γυμνασίου.

```
”NOUN_GEN_MASC_SING_RULES”:
# Ουσιαστικά, Γενικής, Αρσενικό, Ενικός
# σε -ας
”ά”, ”άς”, # Ψαράς
”α”, ”ας”, # Πατέρας, Μήνας, Γείτονας
# σε -ης
”ή”, ”ής”, # Μαθητής
”η”, ”ης”, # Επιβάτης
”ούς”, ”ής”,
# σε -ος
”ίου”, ”ιος”, # Ιούνιος
”ού”, ”ός”, # Βαθμός
”ου”, ”ος”, # Δρόμος
# σε -ες
”έ”, ”ές”, # Καφές
# σε -ούς
”ού”, ”ούς”, # Παππούς
# σε -έας
”έα”, ”έας”, # Κουρέας
# ανώμαλα
”μυός”, ”μυς”,
”πρέσβη”, ”πρέσβης”,
”πρύτανη”, ”πρύτανης”
```

Πίνακας 3: Απόσπασμα των κανόνων του λημματοποιητή της εργασίας

### 3.3.1 Κώδικας

Ο λημματοποιητής της εργασίας ονομάστηκε Lemmatagizer, καθώς ασχολείται με το "tag" της λέξης. Ακολουθούν επιλεγμένα και απλοποιημένα αποσπάσματα κώδικα του Lemmatagizer: Το πρώτο κομμάτι αφορά την εύρεση του κατάλληλου rule table για τη συγκεκριμένη λέξη:

```
# In case of empty POS tag, returns the word as is.
if univ_pos in ('', 'eol', 'space') or str(univ_tag) == '{}':
    return [string.lower()]

# Replacing the old lemmatizer's rules table
self.lookups.remove_table("lemma_rules")
self.lookups.add_table("lemma_rules", custom_rules)

lookup_table = self.lookups.get_table('lemma_lookup', {})
index_table = self.lookups.get_table("lemma_index", {})
exc_table = self.lookups.get_table("lemma_exc", {})
rules_table = self.lookups.get_table("lemma_rules", {})

# Skipping lemmatization for not supported POS
# Forcing noun tables' rules for proper noun and adjective
if univ_pos not in ("noun", "propn", "adj", "verb"):
    return [string.lower()]
elif univ_pos != "verb":
    univ_pos = "noun"

# Remove symbols from words
if any(not char.isalpha() for char in string):
    string = symbol_remover(string)
    if not len(string): return ["-skip-"]

tag = tag_to_rule(univ_pos, str(univ_tag))

if any(char.isdigit() for char in string) or \
        self.is_base_form(tag) or tag == "abbr":
    # Read deaccentizer's docstring
    return deaccentizer(string)

# Using either pos or tag as the same variable
pos_or_tag = tag if rules_table.get(tag) else univ_pos
```

Αφού βρεθεί το κατάλληλο rule table, καλείται η συνάρτηση lemmatize(), όπου γίνεται η εύρεση της σωστής κατάληξης και η αντικατάσταση της υπάρχουσας:

Κλήση της συνάρτησης

```
lemmas = self.lemmatize(
            string,
            index_table.get(univ_pos, {}),
            exc_table.get(univ_pos, {}),
            rules_table.get(pos_or_tag, []),
)
return lemmas
```

Κώδικας της συνάρτησης

```
def lemmatize(self, string, index, exceptions, rules):
    orig = string
    string = string.lower()
    forms = []
    oov_forms = []
    for old, new in rules:
        if string.endswith(old):
            form = string[: len(string) - len(old)] + new
            if not form:
                pass
            elif form in index or not form.isalpha():
                forms.append(form)
            else:
                oov_forms.append(form)
                break
    # Remove duplicates but preserve the ordering of applied "rules"
    forms = list(OrderedDict.fromkeys(forms))
    # Put exceptions at the front of the list, so they get priority.
    for form in exceptions.get(string, []):
        if form not in forms:
            forms.insert(0, form)
    if not forms:
        forms.extend(oov_forms)
    if not forms:
        forms.append(orig)
    return deaccentizer(forms[0])
```

# 4 Μελέτη αλγορίθμων εξαγωγής λέξεων-κλειδιών που χρησιμοποιήθηκαν

## 4.1 Εισαγωγή για το keyword extraction

Η εξαγωγή λέξεων-κλειδιών αποτελεί μια από τις σημαντικότερες μεθόδους απόσπασης πληροφορίας από κείμενα. Μελετώντας τις λέξεις κλειδιά αντί για ολόκληρο το κείμενο, μπορεί κανείς, και ειδικά τα συστήματα ανάλυσης κειμένου, να καταλάβει ταχύτερα τα κύρια σημεία του, καθώς το μέγεθος αυτού μειώνεται στο ελάχιστο. Με αυτόν τον τρόπο επίσης, γίνεται ομαδοποίηση του περιεχομένου, βάσει του θέματος του. (Medelyan, 2014)

Ειδικότερα, στον τομέα των ειδησεογραφικών άρθρων, οι λέξεις-κλειδιά εξοπλίζουν το κείμενο με μια συνοπτική περιγραφή του περιεχομένου του. Αυτό βοηθάει στην κατηγοριοποίηση του άρθρου και στη στοχευμένη εύρεσή τους από μηχανές αναζήτησης. (Vivek, 2018)

Για την εξαγωγή τους, χρησιμοποιούνται λίστες με λέξεις που παράγονται από τα εργαλεία που αναφέρθηκαν στα προηγούμενα κεφάλαια. Με διαφορετικούς τρόπους ο κάθε αλγόριθμος, όπως θα δούμε παρακάτω, μελετάει το κείμενο και υπολογίζει μετρικές για κάθε λέξη, καταλήγοντας έτσι σε μια λίστα των κύριων, πιο σημαντικών λέξεων του κειμένου.

## 4.2 Αλγόριθμοι

Εδώ θα ανελυθούν οι τεχνικές που μελετήθηκαν για την εξαγωγή και επιλογή των λέξεων κλειδιών. Αρχικά, χρήσιμοι και απλοί τρόποι είναι αυτοί που αναφέρθηκαν ως features του κειμένου. Με την κατάλληλη είσοδο, κείμενο περασμένο από προεπεξεργασία, τα αποτελέσματά τους πλησιάζουν τις υπόλοιπες μεθόδους. Ακολουθούν στατιστικές αναλύσεις έως και ολοκληρομένοι αλγόριθμοι.

### 4.2.1 Bag of Words

Αν αγνοήσουμε τον τρόπο παρουσίασής του, αναφέρεται απλά στη συχνότητα εμφάνισης της κάθε λέξεις. Αφαιρώντας τις stop words, ο αριθμός εμφανίσεων μπορεί να περιέχει και μη σημαντικές λέξεις, αλλά αποτελεί την πιο απλή μέθοδο εξαγωγής λέξεων κλειδιών ενός όρου. Χρησιμοποιώντας το BoW αποτέλεσμα που αναλύθηκε στο κεφάλαιο 2.3.1, αν αντικαταστήσουμε τους αριθμούς με τις λέξεις στις οποίες αναφέρονται, προκύπτει η συχνότητα εμφάνισης λέξεων. Υπενθυμίζεται:

'διευθυνση': 0, 'μεταφορες': 1, 'επικοινωνιες': 2, 'δυτικη': 3, 'θεσσαλονικη': 4, 'περιφερεια': 5, 'κεντρικη': 6, 'μακεδονια': 7, 'εξετασεις': 8, 'αποκτηση': 9, 'πιστοποιητικο': 10, 'επαγγελματικη': 11, 'επαρκεια': 12, 'οδικες': 13, 'μεταφορεις': 14, 'ημερα': 15, 'εξεταση': 16, 'δευτερα': 17, 'ιουνιος': 18, '2020': 19, 'αιθουσα': 20, 'ειδικες': 21, 'Καλοχώρι': 22, 'μετρα': 23, 'κορωνοιος': 24, 'αριθμος': 25, 'ωρες': 26, 'προσελευση': 27, 'υποψηφια': 28

(0, 2), (1, 2), (2, 2), (3, 2), (4, 2), (5, 1), (6, 1), (7, 1), (8, 4), (9, 1), (10, 1), (11, 1), (12, 1), (13, 1), (14, 1), (15, 2), (16, 2), (17, 1), (18, 1), (19, 1), (20, 2), (21, 1), (22, 1), (23, 1), (24, 1), (25, 2), (26, 1), (27, 1), (28, 1)

Άρα προκύπτει:

'διευθυνση': 2, 'μεταφορες': 2, 'επικοινωνιες': 2, 'δυτικη': 2, 'θεσσαλονικη': 2, 'περιφερεια': 1, 'κεντρικη': 1, 'μακεδονια': 1, 'εξετασεις': 4, 'αποκτηση': 1, 'πιστοποιητικο': 1, 'επαγγελματικη': 1, 'επαρκεια': 1, 'οδικες': 1, 'μεταφορεις': 1, 'ημερα': 2, 'εξεταση': 2, 'δευτερα': 1, 'ιουνιος': 1, '2020': 1, 'αιθουσα': 2, 'ειδικες': 1, 'Καλοχώρι': 1, 'μετρα': 1, 'κορωνοιος': 1, 'αριθμος': 2, 'ωρες': 1, 'προσελευση': 1, 'υποψηφια': 1

Η φθίνουσα σειρά της βαθμολογίας των λέξεων είναι:

εξετασεις - 4
διευθυνση - 2
μεταφορες - 2
επικοινωνίες - 2
δυτικη - 2
θεσσαλονικη - 2
ημερα - 2
εξεταση - 2
αιθουσα - 2
αριθμος - 2

Φυσικά σε ένα μεγαλύτερο κείμενο, οι τιμές θα ξεχώριζαν περισσότερο.

### 4.2.2 TF-IDF

Αφού εξηγήθηκε η θεωρία στο κεφάλαιο 2.3.2, στο σημείο αυτό θα αναλυθεί ένα παράδειγμα. Για την επεξήγηση του τρόπου λειτουργίας του TF-IDF, θα χρησιμοποιηθεί ο ακόλουθος συλλογισμός, όπου η κάθε πρόταση-γραμμή θα αντιμετωπισθεί ως ξεχωριστό έγγραφο, και το corpus θα αποτελείται από όλα τα έγγραφα που προέκυψαν:

οι άνθρωποι είναι θνητοί
οι ευρωπαίοι είναι άνθρωποι
οι ευρωπαίοι είναι θνητοί
οι ασιάτες είναι άνθρωποι
οι ασιάτες είναι θνητοί
όλοι οι άνθρωποι είναι θνητοί άνθρωποι

Στο πρώτο στάδιο, αυτό του TF, αρχικά καταγράφεται το πόσες φορές εμφανίζεται η κάθε λέξη.

'οι': 1, 'άνθρωποι': 1, 'είναι': 1, 'θνητοί': 1
'οι': 1, 'ευρωπαίοι': 1, 'είναι': 1, 'άνθρωποι': 1
'οι': 1, 'ευρωπαίοι': 1, 'είναι': 1, 'θνητοί': 1
'οι': 1, 'ασιάτες': 1, 'είναι': 1, 'άνθρωποι': 1
'οι': 1, 'ασιάτες': 1, 'είναι': 1, 'θνητοί': 1
'όλοι': 1, 'οι': 1, 'άνθρωποι': 2, 'είναι': 1, 'θνητοί': 1

Οι τιμές αυτές κανονικοποιούνται, διαιρώντας το πλήθος εμφανίσεων με το μήκος του εγγράφου, δηλαδή με το πλήθος των λέξεων που διαθέτει.

'οι': 0.25, 'άνθρωποι': 0.25, 'είναι': 0.25, 'θνητοί': 0.25
'οι': 0.25, 'ευρωπαίοι': 0.25, 'είναι': 0.25, 'άνθρωποι': 0.25
'οι': 0.25, 'ευρωπαίοι': 0.25, 'είναι': 0.25, 'θνητοί': 0.25
'οι': 0.25, 'ασιάτες': 0.25, 'είναι': 0.25, 'άνθρωποι': 0.25
'οι': 0.25, 'ασιάτες': 0.25, 'είναι': 0.25, 'θνητοί': 0.25
'όλοι': 0.17, 'οι': 0.17, 'άνθρωποι': 0.33, 'είναι': 0.17, 'θνητοί': 0.17

Ταυτόχρονα υπολογίζεται το IDF της κάθε λέξης με βάση ολόκληρο το corpus. Το σκορ προκύπτει από τον log $\frac{n}{val}$, όπου n το πλήθος των μοναδικών λέξεων και όπου val ο αριθμός των συνολικών εμφανίσεων της κάθε λέξης.

οι - 0.07, άνθρωποι - 0.15, είναι - 0.07, θνητοί - 0.24, ευρωπαίοι - 0.54, ασιάτες - 0.54, όλοι - 0.85

Τέλος, το TF-IDF σκορ είναι το γινόμενο του TF και του IDF της κάθε λέξης.

'οι': 0.02, 'άνθρωποι': 0.04, 'είναι': 0.02, 'θνητοί': 0.06
'οι': 0.02, 'ευρωπαίοι': 0.14, 'είναι': 0.02, 'άνθρωποι': 0.04
'οι': 0.02, 'ευρωπαίοι': 0.14, 'είναι': 0.02, 'θνητοί': 0.06
'οι': 0.02, 'ασιάτες': 0.14, 'είναι': 0.02, 'άνθρωποι': 0.04
'οι': 0.02, 'ασιάτες': 0.14, 'είναι': 0.02, 'θνητοί': 0.06
'όλοι': 0.14, 'οι': 0.01, 'άνθρωποι': 0.05, 'είναι': 0.01, 'θνητοί': 0.04

Έτσι, όπως θα περίμενε κανείς, παρατηρείται ότι το ψηλότερο σκορ συγκέντρωσαν οι λέξεις που υπάρχουν τις περισσότερες φορές στο κάθε άρθρο, αλλά λιγότερες φορές στα υπόλοιπα. Αντίθετα, λέξεις όπως το "οι" και το "είναι", υπάρχουν στον ίδιο βαθμό σε όλα τα κείμενα και έτσι η σημασία τους κρίνεται μηδαμινή. (Tripathi, 2018)

### 4.2.3 TF-IDF bigrams

Η ίδια τεχνική μπορεί να εφαρμοστεί για την εξαγωγή ζευγαριών λέξεων που έχουν ολοκληρωμένη σημασία όταν είναι συνδεδεμένες, σε σχέση με τις δύο λέξεις ξεχωριστά. Για παράδειγμα, οι λέξεις "διατροφική" και "αξία" ξεχωριστά, δεν θα έβγαζαν το σωστό αποτέλεσμα, ενώ ως ζευγάρι, η σημασία τους αλλάζει και έτσι αντιμετωπίζονται ανάλογα. Αν κάποιο κείμενο μιλάει για αξίες ηθικές, οικονομικές ή διατροφικές, θα ξεχωρίσει η λέξη "αξίες" ενώ δεν προσφέρει την πληροφορία που θα ήταν ικανή αν ήταν συνδεδεμένη με τις "διατροφικές".

Αυτό που διαφέρει στην εφαρμογή του TF-IDF για παραπάνω από μια λέξη (n-άδες λέξεων, n-grams) είναι πως οι λέξεις δεν φτάνει να χωριστούν ανά κενό, αλλά πρέπει να χωριστούν ανά κάθε πιθανό συνεχόμενο ζευγάρι n λέξεων. Ύστερα, τα σκορ θα υπολογιστούν με τον ίδιο ακριβώς τρόπο, μιας και το μόνο που αλλάζει είναι το πλήθος όρων ανά κείμενο, καθώς συμπεριλαμβάνονται και τα ζευγάρια.

### 4.2.4 Textrank

Ο αλγόριθμος αυτός χρησιμοποιεί γράφους για να μελετήσει τις λέξεις ενός κειμένου. Δημοσιεύθηκε σε επιστημονικό άρθρο του University of North Texas (Mihalcea & Tarau, 2004). Υπάρχουν διάφορες εκδοχές του αλγορίθμου σε κώδικα βασισμένες στο άρθρο αυτό. Στα πλαίσια της εργασίας, χρησιμοποιήθηκε ως βάση ο κώδικας και η επεξήγηση του Xu Liang (Liang, 2019).

Ένας από τους πιο διαδεδομένους αλγορίθμους αξιολόγησης κειμένου είναι ο PageRank της Google. Ο αλγόριθμος αυτός αναπτύχθηκε από τους Sergey Brin και Larry Page, οι οποίοι μετά την δημοσίευση του σχετικού επιστημονικού άρθρου, ίδρυσαν την εταιρία. Ο PageRank ήταν ο πρώτος αλγόριθμος που χρησιμοποίησε η εταιρία, και ήταν σε λειτουργία μέχρι και το 2019. Σύμφωνα με τον αλγόριθμο αυτόν, το σύνολο των διαθέσιμων ιστοσελίδων αποτελεί έναν μεγάλο γράφο και κάθε ιστοσελίδα απεικονίζεται από έναν κόμβο. Αν η ιστοσελίδα Α συνδέεται, μέσω υπερσυνδέσμου, με την ιστοσελίδα Β, αυτό απεικονίζεται με μια ακμή. Με την ολοκλήρωση του γράφου κατά αυτόν τον τρόπο, γίνεται η ανάθεση του βάρους της κάθε ιστοσελίδα, κατά τον ακόλουθο τύπο:

$$S(V_i) = (1-d) + d * \sum_{j \in In(V_i)} \frac{1}{|Out(V_j)|} S(V_j), \text{ όπου:}$$

- $S(V_i)$ - το βάρος της ιστοσελίδας i
- $d$ - συντελεστής απόσβεσης, σε περίπτωση που δεν υπάρχουν εξερχόμενες ακμές
- $In(V_i)$ - εισερχόμενες ακμές του i
- $Out(V_j)$ - εξερχόμενες ακμές του j
- $|Out(V_j)|$ - πλήθος εξερχόμενων ακμών

Μετά από την ανάλυση αυτή, είναι πολύ πιο εύκολο να κατανοήσει κανείς τον αλγόριθμο TextRank. Αρκεί να σκεφτεί πως ότι είναι μια ιστοσελίδα για τον PageRank, είναι ένα κείμενο για τον TextRank.

Αφού το κείμενο υποστεί προεπεξεργασία με τις προαναφερόμενες μεθόδους, χρειάζεται να σπάσει σε προτάσεις των φιλτραρισμένων λέξεων. Ακολουθεί το άρθρο-παράδειγμα:

"Από τη Διεύθυνση Μεταφορών και Επικοινωνιών Δυτικής Θεσσαλονίκης της Περιφέρειας Κεντρικής Μακεδονίας ανακοινώνεται ότι οι εξετάσεις απόκτησης πιστοποιητικού επαγγελματικής επάρκειας οδικών μεταφορέων θα διεξαχθούν με πρώτη ημέρα εξέτασης τη Δευτέρα 29 Ιουνίου 2020. Οι εξετάσεις θα πραγματοποιηθούν στην αίθουσα ειδικών εξετάσεων ADR και στην αίθουσα εξέτασης Κ.Ο.Κ. της Διεύθυνσης Μεταφορών και Επικοινωνιών Δυτικής Θεσσαλονίκης (Καλοχώρι). Λόγω των μέτρων του κορωνοιού, ο ακριβής αριθμός των ημερών που θα διαρκέσουν οι εξετάσεις, καθώς και οι ώρες προσέλευσης θα καθοριστούν, αφού προηγουμένως οριστικοποιηθεί ο ακριβής αριθμός των υποψηφίων."

Σπάζοντας το κείμενο σε προτάσεις και απομακρύνοντας τις μη χρήσιμες λέξεις, το αποτέλεσμα είναι το εξής:

('διευθυνση', 'μεταφορες', 'επικοινωνιες', 'δυτικη', 'θεσσαλονικη', 'περιφερεια', 'κεντρικη', 'μακεδονια', 'εξετασεις', 'αποκτηση', 'πιστοποιητικο', 'επαγγελματικη', 'επαρκεια', 'οδικες', 'μεταφορεις', 'ημερα', 'εξεταση', 'δευτερα', 'ιουνιος')
('εξετασεις', 'αιθουσα', 'ειδικες', 'εξετασεις', 'αιθουσα', 'εξεταση', 'Κ.Ο.Κ')
('διευθυνση', 'μεταφορες', 'επικοινωνιες', 'δυτικη', 'θεσσαλονικη', 'καλοχωρι')
('μετρα', 'κορωνοιος', 'αριθμος', 'ημερα', 'εξετασεις', 'ωρες', 'προσελευση', 'αριθμος', 'υποψηφιοι')

Η κάθε λέξη αναπαρίσταται με το γράμμα w:

$$w_1, w_2, w_3, w_4, ..., w_n$$

Ορίζεται ως k το επιθυμητό μέγεθος του κάθε "παραθύρου". Σύμφωνα με το επιστημονικό άρθρο στο οποίο δημοσιεύθηκε ο αλγόριθμος TextRank, το μέγεθος αυτό μπορεί να είναι μεταξύ 2 και 10 λέξεων (Mihalcea & Tarau, 2004).

Τα $(w_1, w_2, ..., w_k)$, $(w_2, w_3, ..., w_{k+1})$, $(w_3, w_4, ..., w_{k+2})$ αποτελούν "παράθυρα". Κάθε ζευγάρι δύο λέξεων θεωρείται πως έχουν μη-κατευθυνόμενες ακμές, δηλαδή χωρίς προσανατολισμό, για παράδειγμα, η ακμή $(w_1, w_2)$ είναι ταυτόσημη με την ακμή $(w_2, w_1)$.

Επιλέγοντας την πρόταση ('μετρα', 'κορωνοιος', 'αριθμος', 'ημερα', 'εξετασεις', 'ωρες', 'προσελευση', 'αριθμος', 'υποψηφιοι') για τη συνέχεια του παραδείγματος, και ορίζοντας το μέγεθος του "παραθύρου", k ίσο με 4, προκύπτουν τα ακόλουθα παράθυρα:

('μετρα', 'κορωνοιος', 'αριθμος', 'ημερα'), ('κορωνοιος', 'αριθμος', 'ημερα', 'εξετασεις'), ('ημερα', 'εξετασεις', 'ωρες', 'προσελευση'), ('εξετασεις', 'ωρες', 'προσελευση', 'αριθμος'), ('ωρες', 'προσελευση', 'αριθμος', 'υποψηφιοι')

Για το πρώτο "παράθυρο", δηλαδή το ('μετρα', 'κορωνοιος', 'αριθμος', 'ημερα'), κάθε ζεύγος λέξεων διαθέτει μη-κατευθυνόμενη ακμή. Προκύπτουν τα ακόλουθα ζευγάρια:

('μετρα', 'κορωνοιος'), ('μετρα', 'αριθμος'), ('μετρα', 'ημερα'), ('κορωνοιος', 'αριθμος'), ('κορωνοιος', 'ημερα'), ('κορωνοιος', 'εξετασεις'), ('αριθμος', 'ημερα')

Σύμφωνα με αυτόν τον γράφο, υπολογίζεται το βάρος της κάθε λέξης-κόμβου, όπως εξηγήθηκε για τον αλγόριθμο PageRank. Εξάγοντας τις δέκα λέξεις με το μεγαλύτερο βάρος, επιστρέφεται το ακόλουθο αποτέλεσμα, σε αύξουσα σειρά η κάθε λέξη με το βάρος της (στρογγυλοποιημένο):

('εξετασεις', 2.59), ('ημερα', 1.82), ('αριθμος', 1.38), ('εξεταση', 1.33), ('δυτικη', 1.24), ('θεσσαλονικη', 1.20), ('επικοινωνιες', 1.11), ('περιφερεια', 1.04), ('ωρες', 1.00), ('αιθουσα', 0.99)

Αν και μια από τις αναμενόμενες λέξεις κλειδιά για το άρθρο αυτό, θα ήταν η λέξη "κορωνοϊός", παρατηρούμε πως η επανάληψη των φράσεων "Διεύθυνση Μεταφορών και Επικοινωνιών Δυτικής Θεσσαλονίκης", "αίθουσα (...) εξέτασης" και "ημέρα (...) εξέτασης" δείχνει στον αλγόριθμο πως το νόημα του κειμένου βασίζεται στις αναφερόμενες λέξεις.

### 4.2.5 Word cloud

Το Word cloud δεν αποτελεί προσδιορισμένο τρόπο επιλογής κορυφαίων λέξεων. Είναι μια δημοφιλής τεχνική οπτικοποίησης συχνών λέξεων όπου οι πιο συχνές ή σημαντικές λέξεις απεικονίζονται πιο έντονες και μεγαλύτερες, μέσα σε ένα σύνολο λέξεων όπου η τελική μορφή του σχήματος θυμίζει σύννεφο, εξού και το όνομα της. Πάνω στην τεχνική αυτή όμως έχουν υλοποιηθεί αλγόριθμοι οι οποίοι ξεχωρίζουν τις λέξεις που θα συμπεριληφθούν στο σχήμα. Θα χρησιμοποιηθεί η εκδοχή της βιβλιοθήκης wordcloud που είναι διαθέσιμη για την Python (Mueller, 2012).

Οι λέξεις περνούν από προεπεξεργασία, γίνεται καταμέτρηση της συχνότητάς τους και κρατώνται οι n πιο συχνές, με το n ορισμένο από τον προγραμματιστή, ενώ η προεπιλογή είναι στις 200 λέξεις. Η συχνότητα κανονικοποιείται και αντιμετωπίζεται ως το βάρος της κάθε λέξης. Τέλος, οι λέξεις μετατρέπονται σε εικόνα, με βάση τις παραμέτρους που έχουν δωθεί. Οι παράμετροι αφορούν τον τύπο της γραμματοσειράς, το μέγιστο μέγεθος, το επιθυμητό σχήμα που θα πρέπει να καλύψουν οι λέξεις, τα χρώματα κ.α. Καθώς όμως η προεπεξεργασία που παρέχει δεν περιλαμβάνει αφαίρεση stop words και λημματοποίηση ή κάποια παρόμοια μέθοδο, τα αποτελέσματα δεν θα είναι ιδανικά, και κρίνεται ιδιαίτερα χρήσιμη η εφαρμογή επιπλέον προεπεξεργασίας.

Χρησιμοποιώντας ξανά το άρθρο-παράδειγμα-καπως με περιορισμό στο πλήθος λέξεων, n=35, παράγεται η ακόλουθη εικόνα:

### 4.2.6 Rake

Με το όνομά του να αποτελεί τα αρχικά του "Rapid Automatic Keyword Extraction", ο Rake δημοσιεύθηκε στο άρθρο "Automatic keyword extraction from individual documents" το 2010 (Stuart Rose, 2010). Η λογική πίσω από αυτόν βασίζεται στο ότι οι λέξεις κλειδιά συχνά περιλαμβάνουν πολλές λέξεις ενώ σπάνια περιλαμβάνουν stopwords ή σημεία στίξης. Επιλέγει τις υποψήφιες λέξεις τοποθετώντας και αναλύωντάς τες σε γράφους.

Στα πλαίσια της εργασίας χρησιμοποιείται η βιβλιοθήκη multi-rake που βασίζεται στον αλγόριθμο του επιστημονικού άρθρου. Αν και η βασική του διαφορά με τον αλγόριθμο του άρθρου είναι η εύκολη παροχή λιστών stop words για διάφορες γλώσσες, επιλέχθηκε γιατί προσφέρει ρυθμίσεις μέσω μεταβλητών για την προσαρμογή των επιλογών του αλγορίθμου ανάλογα με τον επιθυμητό στόχο. Οι μεταβλητές αυτές θέτουν προϋποθέσεις για τις υποψήφιες λέξεις. Μπορούν να καθορίσουν το ελάχιστο πλήθος χαρακτήρων σε μια λέξη, τον μέγιστο αριθμό όρων σε μια λέξη-κλειδί, και την ελάχιστη συχνότητα μιας λέξης. Ως είσοδο παίρνει επίσης μια λίστα με stop words και διαθέτει οριοθέτες φράσεων (phrase delimiters) με βάση τους οποίους σπάει τις προτάσεις. Δηλαδή, το κείμενο χωρίζεται σε σειρές συνεχόμενων λέξεων στα σημεία που υπάρχουν stopwords ή όταν οι λέξεις φτάσουν το προκαθορισμένο όριο πλήθους, ώστε να διατηρούνται αναλοίωτες οι χρήσιμες λέξεις. Υποστηρίζει πως οι συνεχόμενες εμφανίσεις όρων μέσα στις υποψήφιες λέξεις κλειδιά μας επιτρέπουν να εντοπίσουμε τη συνύπαρξη λέξεων χωρίς την ανάγκη για δοκιμή με διαφορετικά μεγέθη παραθύρων.

Ο Rake ξεκινά με το να χωρίζει το κείμενο σε ομάδες υποψήφιων λέξεων-κλειδιών. Εδώ εφαρμόζονται τα φίλτρα που ορίστικαν προηγουμένως. Οι προτεινόμενες τιμές είναι minCharacters = 1, maxWords = 5 και minFrequency = 1. Για κάθε σειρά, κρατείται το πλήθος λέξεων όπου θα χρησιμοποιηθεί αργότερα. Στη συνέχεια, οι λέξεις των σειρών αυτών σπάνε για να μελετηθούν σε επίπεδο λέξης. Εδώ, σημειώνεται η συχνότητα της κάθε λέξης και ο βαθμός του κόμβου της λέξης, όπου υπολογίζεται από το άθροισμα των βαθμών των σειρών στις οποίες η λέξη εμφανίστηκε. Το σκορ της λέξης υπολογίζεται με το πιλίκο του βαθμού της λέξης, προς τη συχνότητά της.

$$word\_score[item] = \frac{word\_degree[item]}{word\_frequency[item]}$$

Το word_degree ευνοεί τους όρους που εμφανίζονται συχνά και είναι σε λέξεις με περισσότερους όρους. Το word_frequency ευνοεί όρους που εμφανίζονται συχνά, χωρίς να υπολογίζει τον αριθμό των υπόλοιπων λέξεων γύρω τους. Η διαίρεσή τους, ευνοεί λέξεις που κυρίζως εμφανίζονται μέσα σε υποψήφιες λέξεις περισσότερων όρων.

Πιο απλά, μια λέξη ξεχωρίζει όταν βρίσκεται μέσα σε μεγάλες σειρές συνεχόμενων λέξεων. Αν η σειρά λέξεων (δηλαδή πάνω από ένας όρος) περιέχει stop words, αυτές δεν αφαιρούνται για να διατηρηθεί το νόημα της. Πριν το κάνει αυτό, ελέγχει αν υπάρχουν και αλλού στο κείμενο οι ίδιες λέξεις με την ίδια stop word και την ίδια σειρά, για να διαβεβαιωθεί πως πρόκειται για ορολογία. Η βαθμολογία μιας σειράς λέξεων, είναι το άθροισμα των βαθμολογιών των όρων που περιέχει, γι' αυτό και ξεχωρίζουν δραματικά σε σχέση με τις μονές λέξεις.

### 4.2.7 Yake

Ο YAKE! είναι ένας αλγόριθμος που υποστηρίζει όλες τις γλώσσες, έχοντας ανάγκη μια λίστα από stop words. Παρουσιάστηκε αρχικά ως online demo εργαλείο που μπορεί να δοκιμάσει κανείς εδώ και είναι επίσης διαθέσιμο ως βιβλιοθήκη στο PyPi.
Το σύστημά του αποτελείται από έξι βασικά κομμάτια:

**Προεπεξεργασία - Text Preprocessing**
Χωρίζει το κείμενο σε ξεχωριστούς όρους, όταν συναντήσει κενό ή ειδικό χαρακτήρα (σύμβολο). Δεν αφαιρεί τις stop words ακόμη.

**Εξαγωγή Χαρακτηριστικών - Feature Extraction**
Εξάγει 5 χαρακτηριστικά για κάθε έναν όρο:

1) Casing, Αποτελεί το "περίβλημα" του όρου.
2) Word Positional, Θεωρεί σημαντικότερες τις λέξεις προς την αρχή του κειμένου.
3) Word Frequency, σημειώνει τη συχνότητα των λέξεων που εμφανίζονται πάνω από μια φορά
4) Word Relatedness to Context, υπολογίζει τις διαφορετικές λέξεις που υπάρχουν πριν και μετά τον κάθε όρο. Με όσες περισσότερες διαφορετικές λέξεις συνορεύει, τόσο λιγότερο σημαντική θεωρείται.
5) Word DifSentence, σημειώνει τη συχνότητα του να βρίσκεται η λέξη σε διαφορετικές προτάσεις. Ανυψώνει όσες λέξεις συχνά υπάρχουν σε διαφορετικές προτάσεις.

Το 3 και το 5 σε συνδιασμό με το 4 θυμίζουν τη λογική του TF-IDF. Όσο πιο συχνά υπάρχουν οι λέξεις σε διαφορετικές προτάσεις, τόσο το καλύτερο, αρκεί να μην επαναλαμβάνονται με διαφορετικές γειτονικές λέξεις (γιατί αυτό θα τις έκανε να μοιάζουν με stop words).

**Ατομική Βαθμολόγηση Όρων - Individual Terms Score**
Στο τρίτο αυτό βήμα, συνδιάζονται τα παραπάνω χαρακτηριστικά για να προκύψει μια μετρική όπου κάθε όρος θα πάρει τον βαθμό του από εκείνη. Στον Yake!, όσο μικρότερη βαθμολογία, τόσο μεγαλύτερη η σημασία του όρου. Το βάρος υπολογίζεται από τον ακόλουθο τύπο:

$$\mathrm{S(kw)} = \frac{\prod_{w \in kw} S(w)}{\mathrm{TF(kw)} * \left(1 + \sum_{w \in kw} S(w)\right)}$$

Το σκορ $S(kw)$ για κάθε λέξη κλειδί προκύπτει από τον πολλαπλασιασμό του σκορ $S(w)$ του πρώτου όρου της υποψήφιας λέξης κλειδί, με τα σκορ των επόμενων όρων της λέξης αυτής. Αυτό διαιρείται με το άθροισμα των σκορ $S(w)$ για να εξισορροπιστούν οι τιμές και να μην επωφεληθούν οι λέξεις με τους περισσότερους όρους. Το αποτέλεσμα αυτό διαιρείται με το $TF(kw)$, δηλαδή το term frequency της λέξης κλειδί, για να υποβαθμίσει τις λιγότερο συχνές υποψήφιες λέξεις.

**Παραγωγή Λίστας Υποψηφίων Λέξεων Κλειδιών - Candidate Keywords List Generation**
Με βάση τα βάρη που προέκυψαν προκύπτει η λίστα με τις υποψήφιες λέξεις κλειδιά με παραπάνω από έναν όρους και μετά την αφαίρεση των stop words, η λίστα με τις υποψήφιες λέξεις-κλειδιά ενός μόνο όρου.

**Αφαίρεση Διπλοεγγραφών - Data Deduplication**
Στο πέμπτο βήμα περιορίζονται οι παρόμοιες λέξεις που προέκυψαν από τα προηγούμενα βήματα. Αυτή η διαδικασία γίνεται με χρήση της απόστασης Levenshtein (Levenshtein, 1966).

**Βαθμολόγηση - Ranking**
Τέλος, το σύστημα εξάγει μια λίστα με λέξεις κλειδιά, με το χαμηλότερο $S(w)$ σκορ να δείχνει την πιο σημαντική λέξη. (Campos et al., 2018β□) (Campos et al., 2018α□), (Campos et al., 2020)

Πριν την εκτέλεσή του μπορούν να προσδιορισθούν οι ακόλουθες παράμετροι:

language, κωδικός χώρας για την εφαρμογή της αντίστοιχης λίστας stop words
max_ngram_size, μέγιστος αριθμός όρων σε μια λέξη κλειδί
deduplication_thresold, σταθερά που ρυθμίζει την συμπεριφορά του πέμπτου βήματος, της αφαίρεσης παρόμοιων λέξεων
deduplication_algo, επιλογή αλγορίθμου που θα χρησιμοποιηθεί στο πέμπτο βήμα
window_size, μέγεθος "παραθύρου"
num_of_keywords, πλήθος λέξεων κλειδιών

Μπορεί κανείς να χρησιμοποιήσει το demo για την εξοικίωσή του με τον τρόπο λειτουργίας του αλγορίθμου, αν και τα αποτελέσματα δεν φιλτράρουν τις stop words.

# 5 Περιβάλλον - Συμπεράσματα

## 5.1 Dataset

Τα δεδομένα που χρησιμοποιήθηκαν για τις δοκιμές, τα παραδείγματα στις επεξηγήσεις αλγορίθμων του 4ου κεφαλαίου και τη μελέτη αποτελεσμάτων αποτελούν dataset που παραχωρήθηκε από την Palo ΕΠΕ. Το dataset αυτό περιέχει πρόσφατα κείμενα από ειδησεογραφικές ιστοσελίδες, blogs και λοιπά άρθρα του ελληνικού διαδικτύου που καλύπτουν ένα ευρύ φάσμα θεμάτων. Επιλέχθηκε και χρησιμοποιήθηκε ώστε οι αλγόριθμοι που εφαρμόστηκαν και ο κώδικας που αναπτύχθηκε για την εργασία να αντιμετωπίσουν όσο το δυνατόν πιο ρεαλιστικές συνθήκες. Στόχος ήταν να μπορούν να ανταπεξέλθουν στην ποικιλία περιεχομένου, ποικιλομορφία κειμένου (διαφορετικό format γραφής ανά πηγή) αλλά και ποσότητα των δεδομένων που θα κληθεί να επεξεργαστεί το πρόγραμμα της εργασίας αυτής.

Χρησιμοποιήθηκε ένα δείγμα του dataset μεγέθους πάνω από 10.000 άρθρων και κειμένων για λόγους ταχύτητας εκτέλεσης κατά την ανάπτυξη του προγράμματος.

## 5.2 Σύγκριση λημματοποιητών

Όπως αναφέρθηκε στην εισαγωγή, καθώς τα δεδομένα δεν περνούσαν από το επίπεδο προεπεξεργασίας που απαιτούσε η εργασία αυτή, κρίθηκε αναγκαία η υλοποίηση μιας πιο προχωρημένης διαδικασίας προεπεξεργασίας. Το κομμάτι αυτό χρειάστηκε τον περισσότερο χρόνο για την υλοποίησή του, καθώς δεν αρκούσε απλά η χρήση ενός εργαλείου, αλλά η εις βάθος ανάλυση του τρόπου λειτουργίας του λημματοποιητή και του spaCy γενικότερα, συγκρίνοντας τη συμπεριφορά του σε διαφορετικές γλώσσες και εκδόσεις. Βασίστηκε πάνω στον ήδη υπάρχων λημματοποιητή που έχει ανεπτυχθεί για την ελληνική γλώσσα, με διαφορά την καλύτερη εκμετάλλευση των εργαλείων του spaCy για πιο ακριβή δεδομένα. Επιλέχθηκε λόγω της εφαρμογής της προεπεξεργασίας σε άρθρα ειδήσεων η μη μετατροπή λέξεων του πληθυντικού στον ενικό, κάτι που συχνά συνηθίζεται στους λημματοποιητές. Αυτό μπορεί να εμφανίζει διπλότυπες λέξεις στα αποτελέσματα, διατηρούνται όμως όροι που ισχύουν μόνο στον πληθυντικό, και μειώνεται η αλλοίωση του περιεχομένου του κειμένου. Λάθη σαφώς και εξακολουθούν να σημειώνονται καθώς ο POS Tagger βασίζεται στο προεκπαιδευμένο μοντέλο για την αναγνώριση του μέρους του λόγου. Αυτό σημαίνει, πως κι ας υπάρχει σωστός κανόνας για κάποια λέξη, αν εκείνη αναγνωριστεί ως διαφορετικό μέρος του λόγου, ή πιο συχνά ως άλλη κλίση ή πρόσωπο, η λημματοποίηση θα επιστρέψει μη ορθό αποτέλεσμα.

Ας δούμε όμως τη διαφορά των δύο εργαλείων, στο άρθρο που χρησιμοποιήθηκε στα προηγούμενα παραδείγματα:

"Από τη Διεύθυνση Μεταφορών και Επικοινωνιών Δυτικής Θεσσαλονίκης της Περιφέρειας Κεντρικής Μακεδονίας ανακοινώνεται ότι οι εξετάσεις απόκτησης πιστοποιητικού επαγγελματικής επάρκειας οδικών μεταφορέων θα διεξαχθούν με πρώτη ημέρα εξέτασης τη Δευτέρα 29 Ιουνίου 2020. Οι εξετάσεις θα πραγματοποιηθούν στην αίθουσα ειδικών εξετάσεων ADR και στην αίθουσα εξέτασης Κ.Ο.Κ. της Διεύθυνσης Μεταφορών και Επικοινωνιών Δυτικής Θεσσαλονίκης Καλοχώρι. Λόγω των μέτρων του κορωνοιού, ο ακριβής αριθμός των ημερών που θα διαρκέσουν οι εξετάσεις, καθώς και οι ώρες προσέλευσης θα καθοριστούν, αφού προηγουμένως οριστικοποιηθεί ο ακριβής αριθμός των υποψηφίων."

Λήμματα λέξεων με τον αρχικό λημματοποιητή:
('διεύθυνση', 'μεταφορή', 'επικοινωνία', 'δυτικός', 'Θεσσαλονίκης', 'περιφέρειας', 'κεντρικός', 'Μακεδονίας', 'εξετάσει', 'απόκτηση', 'πιστοποιητικός', 'επαγγελματικός', 'επάρκεια', 'οδικός', 'μεταφορέο', 'ημέρα', 'εξέταση', 'δευτέρας', 'ιουνίο', 'εξετάσει', 'αίθουσα', 'ειδικός', 'εξετάσεα', 'αίθουσα', 'εξέταση', 'κ.ο.κ', 'διεύθυνση', 'μεταφορή', 'επικοινωνία', 'δυτικός', 'Θεσσαλονίκης', 'Καλοχώρι', 'μέτρο', 'κορωνοιού', 'αριθμός', 'ημερής', 'εξετάσει', 'ώρες', 'προσέλευση', 'αριθμός', 'υποψηφία')

Λήμματα λέξεων με τον λημματοποιητή της εργασίας:
('διευθυνση', 'μεταφορες', 'επικοινωνιες', 'δυτικη', 'θεσσαλονικη', 'περιφερεια', 'κεντρικη', 'μακεδονια', 'εξετασεις', 'αποκτηση', 'πιστοποιητικος', 'επαγγελματικη', 'επαρκεια', 'οδικες', 'μεταφορεις', 'ημερα', 'εξεταση', 'δευτερα', 'ιουνιος', 'εξετασεις', 'αιθουσα', 'ειδικες', 'εξετασεις', 'αιθουσα', 'εξεταση', 'κ.ο.κ', 'διευθυνση', 'μεταφορες', 'επικοινωνιες', 'δυτικη', 'θεσσαλονικη', 'καλοχωρι', 'μετρα', 'κορωνοιος', 'αριθμος', 'ημερα', 'εξετασεις', 'ωρες', 'προσελευση', 'αριθμος', 'υποψηφια')

Σημαντικότερες λέξεις κλειδιά με προεπεξεργασία του:

| Αρχικού λημματοποιητή | Λημματοποιητή της εργασίας |
|---|---|
| εξετάσει 30 | εξετασεις 37 |
| επικοινωνία δυτικός 20 | διευθυνση 28 |
| διεύθυνση 18 | επικοινωνιες δυτικη 27 |
| εξέταση 17 | θεσσαλονικη 21 |
| Θεσσαλονίκης 16 | μεταφορες 17 |
| διευθυνση μεταφορων 13 | εξεταση 17 |
| αίθουσα 13 | ημερα 14 |
| αριθμός 11 | διευθυνση μεταφορων 13 |
| διευθυνση 10 | αιθουσα 11 |
| περιφερειας κεντρικης 10 | περιφερειας κεντρικης 10 |

Παρατηρούμε ότι στην πρώτη περίπτωση οι λέξεις έχουν μαζέψει μικρότερη βαθμολογία καθώς εμφανίζονται σε διαφορετικές μορφές τους.

## 5.3 Σύγκριση αλγορίθμων

Οι αλγόριθμοι που αναφέρθηκαν στο 4ο κεφάλαιο, αν και ακολουθούσαν διαφορετικό τρόπο λειτουργίας, θα παρατηρήσουμε πως γενικά συμπεριφέρονται παρόμοια.

Για την κανονικοποίηση των αποτελεσμάτων, στα πλαίσια που επιτρέπει ο κάθε αλγόριθμος, χρησιμοποιήθηκε μια εμπλουτισμένη λίστα με stop words, που περιλαμβάνει λέξεις και σύμβολα που παρατηρούνταν διάσπαρτα στα κείμενα και χωρίς τη λίστα αυτή θα αντιμετοπίζονταν ως λέξεις. Σε όλους τους αλγορίθμους επίσης εφαρμόστηκε η αφαίρεση των τόνων από τις λέξεις. Όσοι αλγόριθμοι έδιναν τη δυνατότητα, οι λέξεις κλειδιά έχουν περιοριστεί στα μέρη του λόγου: ουσιαστικό, πρωτεύον ουσιαστικό και επίθετο.

Σε μεθόδους όπως η απλή συχνότητα λέξεων του BoW ή οι πράξεις στατιστικής του TF-IDF είναι εύκολη η κατάλληλη προεπεξεργασία του κειμένου, αφού δέχονται λέξεις χωρίς να περιορίζονται ή να εξαρτώνται από κάτι άλλο. Γι' αυτό, η χρήση του TF-IDF επεκτάθηκε έχοντας κοινό λεξιλόγιο μεταξύ των υποψήφιων λέξεων ενός και δύο όρων. Οι πιο σύνθετοι αλγόριθμοι όμως περιορίζουν την ανεκτότητά τους σε προεπεξεργασία, καθώς δίνουν σημασία σε σημεία στίξεις, γειτονικές λέξεις και άλλα.

Μιας και το Textrank απαιτεί λεξιλόγιο και χωρισμό προτάσεων, ενσωματώθηκε χωρίς πρόβλημα η ήδη επεξεργασμένη πληροφορία από το spaCy για να καλύψει τις ανάγκες αυτές.

Το Rake λόγω της ιδιαίτερης λογικής που ακολουθεί, παρουσιάζει ιδανικά αποτελέσματα για λέξεις κλειδιά με πολλαπλούς όρους, καθώς ένα μικρότερο "παράθυρο" θα σήμαινε τον αποκλεισμό των μεγαλύτερων λέξεων κλειδιών. Αντίστοιχα, δεν επιτρέπει επεμβάσεις στην είσοδό του, οπότε τα αποτελέσματα δεν είναι λημματοποιημένα. Συχνά όμως, το Rake προτείνει σωστή λέξη που δεν είχε βρεθεί από τους υπόλοιπους τρόπους. Ορίστηκε "παράθυρο" ίσο με 3 ώστε να επιστρέφει σημαντικότερα αποτελέσματα. Ανταπεξήλθε στις προσδοκίες που περίμενε κανείς με το όνομα αυτό (Rapid), καθώς χρησιμοποίησε μόλις το 7% του συνολικού χρόνου εκτέλεσης.

Το YAKE! εμφανίζει πολύ σωστά αποτελέσματα αλλά για την εξαγωγή λέξεων και με έναν αλλά και με δύο όρους, χρειάστηκε να εκτελεστεί δυο φορές καθώς αν και πολυγλωσσικό, η στα ελληνικά κείμενα έτεινε να δίνει πολύ μεγαλύτερη βαρύτητα στις λέξεις δύο όρων. Όταν το YAKE! εξάγει αποτελέσματα με περισσότερους από έναν όρο, δεν εφαρμόζει προεπεξεργασία για να μην αλλοιώσει το περιεχόμενο. Η λύση δεν ήταν άλλη από το να εκτελεστεί δύο φορές: Την πρώτη, με είσοδο τις προεπεξεργασμένες λέξεις και πλήθος όρων ίσο με 1, για τις λέξεις με έναν όρο, και τη δεύτερη με είσοδο ολόκληρο το κείμενο και πλήθος όρων ίσο με 2. Η πολυπλοκότητα του όμως, σε συνδιασμό με τη διπλή εκτέλεση, οδήγησε μεν σε ικανοποιητικά αποτελέσματα αλλά αποτέλεσε την πιο αργή διαδικασία, με τον χρόνο εκτέλεσης να φτάνει το 77% του συνολικού (στα 10.000 έγγραφα, χρειάστηκε πάνω απο 18 λεπτά με Intel Core i7-3630QM).

Το wordcloud δεν χρησιμοποιήθηκε ως παράγοντας βαρύτητας στις λέξεις κλειδιά, αλλά αξίζει να σημειωθεί η διαφορά του αποτελέσματος όταν γίνεται χρήση λίστας stop words:


Σχήμα 2: Σύγκριση αποτελέσματος wordcloud, πριν και μετά τον προεπεξεργασία

## 5.4 Έξυπνη επιλογή λέξεων

Μετά τη μελέτη και εξέταση διαφορετικών αλγορίθμων, ένιωσα την ανάγκη εξαγωγής μιας μοναδικής λίστας με λέξεις κλειδιά για κάθε κείμενο, συνδιάζοντας τις ανωτέρω πηγές. Υλοποιήθηκε ένα σύστημα ψηφοφορίας με βάση την κατάταξη της κάθε λέξεις σε κάθε έναν από τους αλγορίθμους. Η κλίμακα της μεθόδου βαθμολόγησης είναι από το 1 εώς το πλήθος λέξεων κλειδιών (προκαθορισμένη τιμή ίση με 10, εκτός εάν οι λέξεις του κειμένου μετά την προεπεξεργασία είναι λιγότερες) και κάθε αλγόριθμος βαθμολογεί τη λέξη με την ίδια βαρύτητα.

Σε αυτό το σημείο παρατηρήθηκε πως οι λέξεις κλειδιά με δύο όρους συχνά δεν έπαιρναν τη θέση που άξιζαν, οπότε τέθηκε σε ισχύ ακόμη ένας μηχανισμός πριν το τελικό αποτέλεσμα. Συχνό φαινόμενο ήταν η ύπαρξη του διγράμματος και των δύο μονών όρων ξεχωριστά στην τελική λίστα, με το δίγραμμα να χάνεται μετά την ψηφοφορία. Συνεπώς, στο σημείο αυτό γίνεται έλεγχος για κάθε λέξη κλειδί με δύο όρους, για το αν και οι δύο όροι του υπάρχουν στις λέξεις που ψηφίστηκαν. Στην περίπτωση αυτή, η διαφορά των πόντων της κάθε λέξεις συγκρίνεται με έναν σταθερό συντελεστή (το 2% των συνολικών πόντων όλων των λέξεων) και αν είναι αρκετά μικρή τότε η χαμηλότερα βαθμολογημένη λέξη διαγράφεται και στη θέση της υψηλότερα βαθμολογημένης λέξης μπαίνει το δίγραμμα. Αυτή η λεπτομέρια προκάλεσε σημαντική διαφορά στο τελικό αποτέλεσμα και κάλυψε το κενό των λέξεων με 2 όρους. Ο συντελεστής αυτός προέκυψε μετά από δοκιμές και μπορεί να προσαρμοστεί ανάλογα με τις ανάγκες και τη σημαντικότητα των διγραμμάτων στην κάθε περίπτωση.

Επιπλέον αναπτύχθηκαν μηχανισμοί με σκοπό τη βελτιστοποίηση των αποτελεσμάτων, όπως η ομογενοποίηση των παρόμοιων λέξεων, έτσι ώστε κάθε πιθανή μορφή μιας λέξης να αντικαταστείται από την πιο συνηθισμένη της. Για παράδειγμα, διαφορετικές μορφές αποτελούν η παρουσία ή η έλλειψη τόνου, ή η κατάληξη της λέξης, καταστάσεις που συναντάμε καθώς δεν περνάνε όλοι οι αλγόριθμοι τα αποτελέσματά τους από προεπεξεργασία.

## 5.5 Παραδείγματα εισόδου-εξόδου:

### (α) Παράδειγμα άρθρου - Θέμα: Διακοπές

```
"Περιοχή Ηλείας - Θέλετε να κάνετε κράτηση για τις διακοπές σας; Είτε επιθυμείτε μία ρομαντική απόδραση, ένα οικογενειακό ταξίδι ή διακοπές all-inclusive, τα πακέτα διακοπών
για Περιοχή Ηλείας του TripAdvisor θα διευκολύνουν τον προγραμματισμό, εξοικονομώντας σας παράλληλα χρήματα. Βρείτε το ιδανικό πακέτο διακοπών για: Περιοχή Ηλείας στο
TripAdvisor συγκρίνοντας τιμές ξενοδοχείων και αεροπορικών εισιτηρίων προς και από: Περιοχή Ηλείας. Ταξιδιώτες σαν και εσάς έγραψαν κριτικές και δημοσίευσαν αυθεντικές
φωτογραφίες για Περιοχή Ηλείας ξενοδοχεία. Κάντε κράτηση για τις διακοπές σας (Περιοχή Ηλείας) σήμερα κιόλας! Δείτε τα πακέτα διακοπών για την περιοχή Ηλείας ΕΔΩ"
```

| | WordFreq | TF-IDF (Uni & Bi) | Textrank4kw | YAKE! (Uni) | YAKE! (Bi) | Rake |
|---|---|---|---|---|---|---|
| 1 | (περιοχη, 7) | (περιοχη ηλεια, 0.3802) | (περιοχη, 2.4225) | (περιοχη, 0.0193) | (περιοχη ηλειας, 0.0126) | (δημοσιευσαν αυθεντικες φωτογραφιες, 9.0) |
| 2 | (ηλεια, 7) | (ηλεια, 0.2989) | (διακοπες, 2.2443) | (ηλεια, 0.0271) | (περιοχη, 0.0402) | (ιδανικο πακετο διακοπων, 8.3333) |
| 3 | (διακοπες, 6) | (διακοπες περιοχη, 0.2278) | (ηλεια, 2.1779) | (διακοπες, 0.04) | (ηλειας, 0.0412) | (περιοχη ηλειας ξενοδοχεια, 7.8571) |
| 4 | (κρατηση, 2) | (διακοπες, 0.2197) | (tripadvisor, 1.3877) | (κρατηση, 0.0648) | (ρομαντικη αποδραση, 0.0701) | (περιοχη ηλειας σημερα, 7.8571) |
| 5 | (πακετα, 2) | (περιοχη, 0.183) | (ταξιδι, 0.9976) | (πακετα, 0.0861) | (παραλληλα χρηματα, 0.0701) | (περιοχη ηλειας εδω, 7.8571) |
| 6 | (tripadvisor, 2) | (tripadvisor, 0.1205) | (all-inclusive, 0.96) | (tripadvisor, 0.0861) | (θελετε, 0.0781) | (περιοχη ηλειας, 4.8571) |
| 7 | (ρομαντικη, 1) | (ηλεια κρατηση, 0.1205) | (αυθεντικες, 0.9512) | (εδω, 0.1083) | (οικογενειακο ταξιδι, 0.0968) | (πακετα διακοπων, 4.3333) |
| 8 | (αποδραση, 1) | (κρατηση διακοπες, 0.1205) | (φωτογραφιες, 0.9512) | (ρομαντικη, 0.1536) | (ηλειας εδω, 0.1212) | (κανετε κρατηση, 4.0) |
| 9 | (οικογενειακο, 1) | (ηλεια tripadvisor, 0.1205) | (πακετα, 0.9447) | (αποδραση, 0.1536) | (πακετα διακοπων, 0.1328) | (ρομαντικη αποδραση, 4.0) |
| 10 | (ταξιδι, 1) | (πακετα διακοπες, 0.1166) | (συγκρινοντα, 0.9246) | (οικογενειακο, 0.1536) | (διακοπων, 0.1353) | (οικογενειακο ταξιδι, 4.0) |

```
Info: Term Merge: αποδραση + ρομαντικη => ρομαντικη αποδραση
Info: Term Merge: οικογενειακο + ταξιδι => οικογενειακο ταξιδι


Final List of Keywords
περιοχη 45
ηλεια 35
διακοπες 32
tripadvisor 22
περιοχη ηλειας 15
κρατηση 14
πακετα 14
περιοχη ηλεια 10
δημοσιευσαν αυθεντικες φωτογραφιες 10
ιδανικο πακετο διακοπων 9
```

### (β) Παράδειγμα άρθρου - Θέμα: Διατροφή

```
"Η μοναστηριακή κουζίνα είναι γεμάτη εκπληκτικές συνταγές. Οι μοναχοί χρησιμοποιώντας απλά υλικά δημιουργούν ξεχωριστές γεύσεις, πλούσιες σε θρεπτικά συστατικά.
Αξίζει σίγουρα να "τολμήσετε" τους: κάβουρες γιαχνί! Για να φτιάξουμε κάβουρες γιαχνί θα χρειαστούμε: 2 κάβουρες μεγάλους, 2 κρεμμύδια ψιλοκομμένα,
4 σκελίδες σκόρδο ψιλοκομμένο.... Διαβάστε αναλυτικά τη συνταγή στο epirutoday.com"
```

| | WordFreq | TF-IDF (Uni & Bi) | Textrank4kw | YAKE! (Uni) | YAKE! (Bi) | Rake |
|---|---|---|---|---|---|---|
| 1 | (καβουρες, 3) | (καβουρες, 0.3658) | (γευσεις, 1.3187) | (γιαχνι, 0.0939) | (εκπληκτικες συνταγες, 0.0925) | (γεματη εκπληκτικες συνταγες, 9.0) |
| 2 | (γιαχνι, 2) | (γιαχνι, 0.2519) | (καβουρες, 1.2507) | (καβουρες, 0.1031) | (μοναστηριακη κουζινα, 0.1573) | (φτιαξουμε καβουρες γιαχνι, 8.0) |
| 3 | (μοναστηριακη, 1) | (καβουρες γιαχνι, 0.2519) | (ξεχωριστες, 1.1387) | (μοναστηριακη, 0.1398) | (γεματη εκπληκτικες, 0.1573) | (καβουρες γιαχνι, 5.0) |
| 4 | (κουζινα, 1) | (γιαχνι καβουρες, 0.2519) | (πλουσιες, 1.1387) | (συνταγη, 0.1398) | (συνταγες, 0.2296) | (μοναστηριακη κουζινα, 4.0) |
| 5 | (γεματη, 1) | (συνταγη, 0.1795) | (μεγαλοι, 1.0865) | (κουζινα, 0.2416) | (καβουρες, 0.2595) | (θρεπτικα συστατικα, 4.0) |
| 6 | (εκπληκτικες, 1) | (μοναστηριακη, 0.1328) | (κρεμμυδια, 1.0865) | (γεματη, 0.2416) | (γιαχνι, 0.3419) | (αξιζει σιγουρα, 4.0) |
| 7 | (συνταγες, 1) | (κρεμμυδια, 0.1328) | (κουζινα, 1.0815) | (εκπληκτικες, 0.2416) | (μοναστηριακη, 0.3687) | (2 κρεμμυδια ψιλοκομμενα, 4.0) |
| 8 | (μοναχοι, 1) | (μοναστηριακη κουζινα, 0.1328) | (γεματη, 1.0815) | (συνταγες, 0.2416) | (κουζινα, 0.3687) | (διαβαστε αναλυτικα, 4.0) |
| 9 | (υλικα, 1) | (κουζινα γεματη, 0.1328) | (εκπληκτικες, 1.0815) | (μοναχοι, 0.2416) | (γεματη, 0.3687) | (πλουσιες, 1.0) |
| 10 | (ξεχωριστες, 1) | (γεματη εκπληκτικες, 0.1328) | (υλικα, 0.9487) | (υλικα, 0.2416) | (εκπληκτικες, 0.3687) | (τολμησετε, 1.0) |

```
Info: Term Merge: κουζινα + μοναστηριακη => μοναστηριακη κουζινα
Info: Term Merge: εκπληκτικες + γεματη => γεματη εκπληκτικες
Info: Term Merge: υλικα + μοναχοι => μοναχοι υλικα
Info: Term Merge: ξεχωριστες + γευσεις => ξεχωριστες γευσεις


Final List of Keywords
καβουρες 44
γιαχνι 33
μοναστηριακη κουζινα 25
καβουρες γιαχνι 16
γεματη εκπληκτικες 16
συνταγες 14
συνταγη 13
εκπληκτικες συνταγες 10
γεματη εκπληκτικες συνταγες 10
ξεχωριστες γευσεις 10
```

### (γ) Παράδειγμα άρθρου - Θέμα: Ταξίδια

```
"Αν οι Locomondo είχαν κάνει ένα ταξιδάκι προς Κεφαλονιά μερικά και είχαν βουτήξει στα νερά της πιο παγωμένης θάλασσας στην Ελλάδα, πιθανότητα να μην είχαν όρεξη να
γράψουν ποτέ το "Δεν κάνει κρύο στην Ελλάδα". Μπορούμε να σας διαβεβαιώσουμε ότι σε αυτόν τον μικρό παράδεισο, σε αυτή την ανειρική παραλία με τα παγωμένα νερά,
πιο εύκολα τραγουδάς το "Δεν κάνει ζέστη, στην Ελλάδα, ζέστη δεν έκανε ποτέ". Ανάμεσα στα χαρτιά Χαβριάτα και Χαβδάτα στην Παλική βρίσκεται ένας πραγματικός παράδεισος.
Οι κάτοικοι του Ληξουρίου (ανεξάρτητο κρατίδιο που δεν έχει την παραμικρή σχέση με την Κεφαλονιά) γνωρίζουν αυτή τη μικρή, αλλά υπέροχη παραλία με τα άσπρα βότσαλα,
τα περίφημα Λαγκαδάκια. Όσοι την έχουν επισκεφτεί έστω και μια φορά γνωρίζουν ότι η συγκεκριμένη παραλία, με τα κρυστάλλινα νερά έχει μια μικρή ιδιαιτερότητα.
Δεν ζεσταίνει ποτέ... Ακόμα και αν η θερμοκρασία έξω είναι 45 βαθμοί Κελσίου, το νερό στα Λαγκαδάκια είναι τόσο παγωμένο, που νομίζεις ότι μόλις έχει βγει από το ψυγείο.
Τα υπόγεια ρεύματα που υπάρχουν στο συγκεκριμένο μέρος δεν αφήνουν ποτέ τη θάλασσα να ζεστάνει γι' αυτό όποια μέρα του χρόνου και αν πάτε δεν θα καταλάβετε διαφορά.
Αυτός είναι και ένας απ' τους λόγους που η συγκεκριμένη παραλία μπαίνει στη λίστα με εκείνες που οι ντόπιοι σου λένε ότι πρέπει να επισκεφτείς αν ο δρόμος σου σε βγάλει
στην Κεφαλονιά. Μπορεί να μην έχει τη φήμη του Μύρτου ή την άγρια ομορφιά των Πεταναών, αλλά είναι το μικρό κρυμμένο μυστικό της περιοχής που πρέπει να ανακαλύψεις.
Και τα υπέροχα, κρυστάλλινα νερά της δεν είναι ο μοναδικός λόγος για να την αναζητήσεις. Αν πάρεις μαζί σου μια μάσκα και καλυμπήσεις στα Λαγκαδάκια θα δεις κάτω από την
επιφάνεια της θάλασσας ένα ασύλληπτης ομορφιάς θέαμα που θα σου κόψει την ανάσα. Οι... θαλασσόλυκοι του νησιού θα σου πουν ότι αν το πάθος σου είναι οι καταδύσεις,
τότε είσαι στο σωστό μέρος. Στα Λαγκαδάκια θα γνωρίσεις έναν νέο κόσμο που θα θυμάσαι για μια ζωή. Στα Λαγκαδάκια δε θα βρεις μια οργανωμένη παραλία. Θα βρεις έναν
ανόθευτης ομορφιάς παράδεισο που αν τον επισκεφτείς μια φορά, θα σε καλεί πίσω για πάντα.
ΚΑΛΟΚΑΙΡΙ 2020 - ΠΡΟΤΑΣΕΙΣ ΔΙΑΚΟΠΩΝ
Γαληνεύει την ψυχή: Το νησί-έρωτας που αν του δώσεις μια ευκαιρία, δεν θα το αλλάξεις ποτέ ξανά (pics) Ονειρικά νερά, φαγητό που δεν θα ξεχάσεις ποτέ. Το αυτοκίνητο σου
είναι άχρηστο: Το νησί-outsider που σου προτείνουν οι πολύ "φαγμένοι" και εδώ που τα λέμε, πολύ καλά κάνουν.
Νησί-ηφαίστειο: Τα καταγάλανα νερά της Νισύρου που θα κάνουν να ... ετοιμάσεις βαλίτσες Ένα νησί που δεν έχει κερδίσει την αναγνώριση που του αξίζει.
Εδώ ο χρόνος σταματά: Στο νησί με τις 18 παραλίες που μπορείς να πας ποδαράτο, θα κάνεις τις καλύτερες διακοπές της ζωής σου (pics) Εδώ ο χρόνος σταματά.
Η ψυχή σου ηρεμεί και το σώμα σου προσπαθεί να αντέξει την ομορφιά που συναντά γύρω του. Ιθάκη: Το ζευγαρονήσι με τη \"μυστική\" παραλία που κάνει τους πάντες να
ερωτευτούν από την αρχή Τι να (πρωτο)πείς για τις ομορφιές αυτού του μέρους; ΚΑΤΕΒΑΣΤΕ ΤΟ ΝΕΟ APP ΤΟΥ ΕΘΝΟΥΣ"
```

| | WordFreq | TF-IDF (Uni & Bi) | Textrank4kw | YAKE! (Uni) | YAKE! (Bi) | Rake |
|---|---|---|---|---|---|---|
| 1 | (παραλια, 7) | (παραλια, 0.1006) | (παραλια, 4.0883) | (παραλια, 0.0476) | (ελλαδα, 0.0519) | (ληξουριου ανεξαρτητο κρατιδιο, 9.0) |
| 2 | (ελλαδα, 4) | (λαγκαδακια, 0.0681) | (νερα, 2.3781) | (νερα, 0.0577) | (ελλαδας, 0.09) | (ασυλληπτης ομορφιας θεαμα, 9.0) |
| 3 | (νερα, 4) | (νερα, 0.0669) | (ομορφια, 2.1393) | (ελλαδα, 0.0662) | (παραλια, 0.0979) | (μικρο κρυμμενο μυστικο, 8.5) |
| 4 | (κεφαλονια, 3) | (κεφαλονια, 0.0591) | (θαλασσα, 1.999) | (συγκεκριμενη, 0.0726) | (λαγκαδακια, 0.1041) | (ανοθευτης ομορφιας παραδεισο, 8.5) |
| 5 | (λαγκαδακια, 3) | (ομορφια, 0.049) | (κεφαλονια, 1.9969) | (κρυσταλλινα, 0.0726) | (κεφαλονια, 0.1356) | (συγκεκριμενη παραλια μπαινει, 7.5) |
| 6 | (ομορφια, 3) | (συγκεκριμενη παραλια, 0.0476) | (λαγκαδακια, 1.8736) | (κεφαλονια, 0.0755) | (νερα, 0.1453) | (μικρο παραδεισο, 5.0) |
| 7 | (παραδεισο, 2) | (επισκεφτεις, 0.0454) | (ελλαδα, 1.7198) | (λαγκαδακια, 0.0755) | (πιανει ζεστη, 0.1478) | (συγκεκριμενη παραλια, 4.5) |
| 8 | (μικρη, 2) | (κρυσταλλινα, 0.044) | (μερος, 1.692) | (ομορφια, 0.0755) | (ζεστη, 0.1609) | (κρυα παραλια, 4.0) |
| 9 | (φορα, 2) | (κρυσταλλινα νερα, 0.044) | (μικρη, 1.4409) | (φορα, 0.0822) | (κανει, 0.1795) | (ονειρικη παραλια, 4.0) |
| 10 | (συγκεκριμενη, 2) | (παραδεισο, 0.0415) | (κρυσταλλινα, 1.3252) | (παραδεισο, 0.0922) | (βαθμους, 0.2067) | (ευκολα τραγουδας, 4.0) |

```
Final List of Keywords
παραλια 48
νερα 39
ελλαδα 31
λαγκαδακια 31
κεφαλονια 31
ομορφια 22
ληξουριου ανεξαρτητο κρατιδιο 10
κρυσταλλινα 10
ελλαδας 9
ασυλληπτης ομορφιας θεαμα 9
```

## 5.6 Έλεγχος ποιότητας αποτελεσμάτων

Με σκοπό εύρεσης μιας μετρικής αξιολόγησης των λέξεων κλειδιών που εξάγει το πρόγραμμα, ζητήθηκε από τρία άτομα να εξάγουν λέξεις κλειδιά από 50 τυχαία άρθρα. Ύστερα, τα αποτελέσματα του προγράμματος συγκρίθηκαν με εκείνα που εξήχθηκαν χειροκίνητα από τους ανθρώπους. Η σύγκριση έγινε με την τελική λίστα και όχι με τον κάθε αλγόριθμο ξεχωριστά, καθώς τα αποτελέσματα θα ήταν άνισα για τους αλγορίθμους που εξάγουν μόνο μονογράμματα ή μόνο διγράμματα. Από την σύγκριση αυτή προέκυψε πως το πρόγραμμα εξάγει λέξεις κλειδιά με ακρίβεια 67,8% ως προς τις λέξεις που θα εξήγαγαν άνθρωποι.

## 5.7 Μελλοντικοί στόχοι

Ξεκινώντας από το πρώτο κομμάτι της εργασίας, η πορεία του για το μέλλον είναι η προσαρμογή του λημματοποιητή στα μέτρα του spaCy κατόπιν επικοινωνίας με τους συντονιστές. Αυτό, επειδή είναι ο πρώτος λημματοποιητής που εξετάζει τόσο βαθιά τη γραμματική μιας λέξεις και προσφέρει κανόνες ειδικούς για την κάθε ομάδα λέξεων! Δευτερεύον στόχος είναι η επανεκπαίδευση ενός ελληνικού μοντέλου για ακριβέστερη αναγνώρηση από τον POS Tagger, ώστε ο λημματοποιητής να μπορεί να φτάσει το όριο των δυνατοτήτων του όσον αφορά την ορθότητα του αποτελέσματος.

Στο δεύτερο κομμάτι, την εξαγωγή λέξεων-κλειδών που αποτελούσε και το αρχικό θέμα της εργασίας, είναι κυρίως η επαλήθευση των αποτελεσμάτων. Μιας και ολοκληρώθηκε με επιτυχία η unsupervised εξαγωγή λέξεων κλειδιών, θα στηθεί ένα σύστημα διαφορετικής προσέγγισης, με χρήση Deep Learning, με σκοπό έλεγχο της ποιότητας των αποτελεσμάτων.

# Αναφορές


Bagola, D. H. (2002). *List of stopwords.*

Bird Steven, E. L., & Klein, E. (2001). *Natural language toolkit.*

Campos, R., Mangaravite, V., Pasquali, A., Jorge, A., Nunes, C., & Jatowt, A. (2020). Yake! keyword extraction from single documents using multiple local features. *Information Sciences*, *509*, 257--289.

Campos, R., Mangaravite, V., Pasquali, A., Jorge, A. M., Nunes, C., & Jatowt, A. (2018α□). A text feature based automatic keyword extraction method for single documents. In *European conference on information retrieval* (pp. 684--691).

Campos, R., Mangaravite, V., Pasquali, A., Jorge, A. M., Nunes, C., & Jatowt, A. (2018β□). Yake! collection-independent automatic keyword extractor. In *European conference on information retrieval* (pp. 806--810).

Honnibal, M. (2015). *Introducing spacy.*

Johnson, K. P., Burns, P., Stewart, J., & Cook, T. (2014--2020). *Cltk: The classical language toolkit.* Retrieved from https://github.com/cltk/cltk

Koutropoulou, T. (2018). *Αυτοματοποιημένη εξαγωγή λέξεων-κλειδιών και ευρετηριοποίηση: Αλγόριθμοι και υλοποιήσεις* (Unpublished doctoral dissertation).

Kozaczko, D. (2018). 8 best python nlp libraries.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady* (Vol. 10, pp. 707--710).

Liang, X. (2019). Understand textrank for keyword extraction by python. *Towards Data Science*.

Medelyan, A. (2014). *Nlp keyword extraction tutorial with rake and maui.* online.

Mihalcea, R., & Tarau, P. (2004). Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing* (pp. 404--411).

Mueller, A. (2012). A wordcloud in python. *peekaboo-vision*.

Padmanabhan, A. (2019). Natural language toolkit.

Stuart Rose, N. C. W. C., Dave Engel. (2010). Automatic keyword extraction from individual documents.

Tripathi, M. (2018). How to process textual data using tf-idf in python. *Edição: free-CodeCamp*.

Vivek, S. (2018). Automated keyword extraction from articles using nlp. *Medium https://medium.com/analytics-vidhya/automated-keyword-extraction-from-articles-using-nlp-bfd864f41b34*.