ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Ταξινόμηση κειμένων με χρήση γράφων λέξεων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Βούλγαρη Σωτήρη

Επιβλέπων : Γιώργος Στάμου
Αναπληρωτής Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2019

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΈΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Ταξινόμηση κειμένων με χρήση γράφων λέξεων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Βούλγαρη Σωτήρη

Επιβλέπων : Γιώργος Στάμου
Αναπληρωτής Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 10η Ιουλίου 2019.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Γιώργος Στάμου Αναπληρωτής Καθηγητής Ε.Μ.Π	Νικόλαος Παπασπύρου Αναπληρωτής Καθηγητής Ε.Μ.Π.	Ανδρέας-Γεώργιος Σταφυλοπάτης Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2019

(Υπογραφή)

.....

Βούλγαρης Σωτήρης

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Οι γράφοι λέξεων αναπαριστούν ένα κείμενο ως ένα γράφο, οι κόμβοι του οποίου είναι οι ξεχωριστοί όροι του κειμένου και οι ακμές συμβολίζουν τη συνύπαρξη δύο όρων σε ένα κινούμενο παράθυρο. Το μοντέλο εκμεταλλεύεται τη σχέση που έχουν μεταξύ τους οι κοντινοί όροι και τη σειρά τους, για να αποδώσει κατάλληλο βάρος στους όρους του κειμένου, το οποίο προκύπτει από το βαθμό που έχει ο αντίστοιχος κόμβος στο γράφο. Το βάρος του όρου μπορεί να αντικαταστήσει τη συχνότητα στη διανυσματική αναπαράσταση TF-IDF, οπότε και προκύπτει το TW-IDF, το οποίο μπορεί να χρησιμοποιηθεί μεταξύ άλλων για την ταξινόμηση κειμένων.

Στην παρούσα εργασία, στόχος είναι η μελέτη και η βελτίωση του μοντέλου γράφων λέξεων στην ταξινόμηση κειμένου. Για το σκοπό αυτό προτείνονται διάφορες τροποποιήσεις του μοντέλου, οι οποίες αφορούν τόσο την προεπεξεργασία του κειμένου όσο και την κατασκευή του γράφου. Πιο συγκεκριμένα, οι μέθοδοι coreference resolution και collocation detection έχουν στόχο τη δημιουργία πιο αντιπροσωπευτικών ακμών και κόμβων αντίστοιχα, μέσω κατάλληλης προεπεξεργασίας του κειμένου. Έπειτα, εξετάστηκε η χρήση της απόστασης ομοιότητας των word embeddings των όρων για τα βάρη των ακμών. Παράλληλα, εντοπίστηκε μια αδυναμία των γράφων λέξεων να δώσουν κατάλληλο βάρος στους όρους που βρίσκονται στα άκρα του κειμένου και για το λόγο αυτό αναπτύχθηκαν οι μέθοδοι ενίσχυσης του βάρους των κόμβων, Rebase και Boost. Η μέθοδος Rebase θέτει ένα κάτω όριο στο βάρος που επιτρέπεται να έχει κάθε όρος , ενώ η Boost τροποποιεί επιλεκτικά τα βάρη μόνο των προβληματικών όρων. Μια άλλη τροποποίηση που εξετάστηκε είναι το μεταβλητό μήκος παραθύρου, στην οποία κάθε όρος έχει το δικό του μέγεθος παραθύρου. Το μέγεθος του παραθύρου καθορίζει το πλήθος των συνδέσεων που έχει ένας κόμβος και ως αποτέλεσμα τη σημασία του αντίστοιχου όρου, οπότε η αλλαγή του μπορεί να επηρεάσει σημαντικά το βάρος που αποδίδει το μοντέλο σε κάθε όρο. Τέλος, προτείνεται η χρήση ensembles γράφων λέξεων. για να εκμεταλλευτούμε τις διάφορες επιλογές που υπάρχουν για την κατασκευή των γράφων και να βελτιώσουμε περαιτέρω την απόδοση της ταξινόμησης.

Η χρησιμότητα των μεθόδων αξιολογείται σε δύο διαφορετικές συλλογές κειμένων, απ΄ όπου προκύπτουν χρήσιμα συμπεράσματα για το μοντέλο γράφου λέξεων, ενώ προτείνονται και κατευθύνσεις για μελλοντική επέκταση και βελτίωση των προτεινόμενων τροποποιήσεων.

Λέξεις κλειδιά
γραφός λέξεων, graph of words, ταξινόμηση κειμένου, coreference resolution, collocation detection, ενίσχυση κόμβων, word embeddings, μεταβλητό μέγεθος παραθύρου, ensembles

Abstract

Graph of words (GoWs) represent a textual document as a graph whose vertices are the unique terms and the edges represent co-occurrence between the terms within a fixed size sliding window. GoWs take into account the relationship that exists between the terms, their order and distance inside the text and uses the degree of a node to assign weight to the corresponding term. The weight of a term can replace the frequency in TF-IDF, which results in TW-IDF, that can be used for text classification.

The scope of this diploma thesis is to examine and improve the GoWs model for the task of text classification. As a result, we propose several modifications for the preprocessing of the text and the construction of the graph. Coreference resolution and collocation detection are used to produce more suitable edges and nodes accordingly. Furthermore, we examined the use of the similarity distance of the terms word embeddings to assign weights to the edges. Regarding a problem in the misrepresentation of the term weight in the edges of the document, we proposed two node reinforcement methods, Rebase and Boost. Rebase defines a lower limit for the term weights and Boost modifies only the term weights that are misrepresented. We also examined the possibility for each term to have its own variable window size. The amount of connections that a node has is proportional to the window size, which means that a bigger size can significantly change the importance of a term. Last but not least, we used ensembles of GoWs to take advantage of the many options we have for the graph creation, to further improve the classification performance.

For the evaluation of the methods we used two different collections of documents and future research is suggested for the enhancement of the proposed methods and the GoWs model in general.

Key Words
graph of words, GoW, text classification, coreference resolution, collocation detection, node reinforcement, word embeddings, variable window size, ensemble

Ευχαριστίες
Αρχικά, θα ήθελα να ευχαριστήσω τους φίλους μου τόσο μέσα όσο και έξω από τη σχολή που με βοήθησαν σε αυτό το σημαντικό βήμα της ζωής μου και με συντρόφευσαν όλα αυτά τα χρόνια. Ευχαριστώ τον επιβλέποντα καθηγητή κύριο Γιώργο Στάμου για την εμπιστοσύνη που μου έδειξε και τον υποψήφιο διδάκτορα Αλέξη Μανδαλιό, ο οποίος συνέβαλλε τα μέγιστα για την ολοκλήρωση της διπλωματικής και ήταν πάντα πρόθυμος να με βοηθήσει και να συζητήσει μαζί μου. Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια για τη συνεχή στήριξη που μου έχει προσφέρει τόσα χρόνια.

Βούλγαρης Σωτήρης
Αθήνα, 9 Ιουλίου 2019

Εικόνες

Πίνακες

Κεφάλαιο 1

Εισαγωγή

1.1 Κίνητρο

Η αναπαράσταση κειμένου βρίσκεται στον πυρήνα διαφόρων προβλημάτων του τομέα επεξεργασίας φυσικής γλώσσας, αφού για να επεξεργαστεί κανείς ένα κείμενο ή μια συλλογή κειμένων θα πρέπει πρώτα να το μετατρέψει σε μια κατάλληλη μορφή, χωρίς ωστόσο να χαθούν οι πληροφορίες που περιέχει το κείμενο. Τυπικά, τα διάφορα μοντέλα αναπαράστασης βασίζονται στη συχνότητα των λέξεων ή συλλαβών, για να εξάγουν χαρακτηριστικά από το κείμενο και να καταλήξουν σε μια διανυσματική αναπαράσταση του κείμενου, όπου κάθε όρος αντιστοιχίζεται σε ένα βάρος που αντιπροσωπεύει τη σημασία που έχει.

Οι γράφοι λέξεων (Graphs of Words ή GoW), όπως ορίστηκαν για πρώτη φορά στο [1], αμφισβητούν την υπόθεση ανεξαρτησίας των λέξεων, ώστε να λάβουν υπόψη την εξάρτηση των λέξεων μεταξύ τους , τη σειρά τους μέσα στο κείμενο και τη μεταξύ τους απόσταση. Η υπόθεση ανεξαρτησίας των λέξεων σημαίνει ότι οι λέξεις του κειμένου αντιμετωπίζονται ανεξάρτητα η μια από την άλλη, χωρίς να ενδιαφέρει η μεταξύ τους σχέση, όπως υποθέτουν τα μοντέλα που βασίζονται στη συχνότητα. Τα κείμενα όμως δεν αποτελούν απλά μια συλλογή λέξεων, αλλά αποτελούνται από προτάσεις, η μορφή των οποίων καθορίζεται από το συντακτικό και η σειρά των λέξεων παίζει πολύ σημαντικό ρόλο για την κατανόηση τους, με συνέπεια να αναπτύσσονται σύνθετες σχέσεις εξαρτήσεων μεταξύ των λέξεων ενός κειμένου. Για το λόγο αυτό, το μοντέλο γράφων λέξεων επιχειρεί να αναπαραστήσει το κείμενο ως ένα γράφο, οι κόμβοι του οποίου θα είναι οι λέξεις και οι ακμές θα καθορίζονται βάση της απόστασης των λέξεων μέσα στο κείμενο. Διαισθητικά, αναμένουμε ότι λέξεις που βρίσκονται κοντά μέσα στο κείμενο έχουν κάποια σχέση και αυτήν ακριβώς τη σύνδεση προσπαθεί να συλλάβει ο γράφος λέξεων. Επιπλέον, η εξαγωγή χαρακτηριστικών δεν βασίζεται πλέον στο κείμενο, αλλά σε μια ενδιάμεση αναπαράσταση, αυτή του γράφου.

Ένα από τα προβλήματα της επεξεργασίας φυσική γλώσσας που μπορεί να επωφεληθεί από μια καλύτερη αναπαράσταση κειμένου, είναι αυτό της ταξινόμησης κειμένου, δηλαδή η ανάθεση ενός κειμένου μιας συλλογής σε μια ή περισσότερες κλάσεις, με αυτόματο τρόπο. Η ταξινόμηση κειμένου είναι ένα από τα πιο παραδοσιακά και καλά μελετημένα προβλήματα, ενώ πρόσφατα με την εισαγωγή τεχνικών βαθιάς μηχανικής μάθησης, η επίδοση των μοντέλων ταξινόμησης έχει φτάσει σε νέα ύψη. Οι εφαρμογές της ταξινόμησης είναι αναρίθμητες, αφού συνήθως για κάθε κείμενο υπάρχει η ανάγκη κατηγοριοποίησης.

Ενδεικτικά αναφέρονται η ταξινόμηση ειδήσεων, η αναγνώριση ανεπιθύμητων μηνυμάτων, η ταξινόμηση σελίδων του διαδικτύου και η ανάλυση συναισθήματος μηνυμάτων.

Συνεπώς, για να διερευνηθεί η απόδοση των γράφων λέξεων στην αναπαράσταση κειμένου, μπορούμε να χρησιμοποιήσουμε το πρόβλημα της ταξινόμησης κειμένου για την αξιολόγηση. Σκοπός είναι η μελέτη των χαρακτηριστικών των γράφων λέξεων και η ανάπτυξη μεθόδων για την βελτίωση τους και όχι η καλύτερη δυνατή επίδοση στην ταξινόμηση, η οποία όμως προκύπτει έμμεσα.

1.2 Συνεισφορά

Στο πλαίσιο της εργασίας υλοποιήθηκε, μελετήθηκε και τροποποιήθηκε το μοντέλο γράφων λέξεων και εφαρμόστηκε στο πρόβλημα της ταξινόμησης κειμένου. Ο κύριος στόχος των μεθόδων που αναπτύχθηκαν είναι η καταλληλότερη αναπαράσταση του κειμένου από τους γράφους λέξεων. Αρχικά, αναγνωρίστηκε η ανάγκη για καλύτερη προεπεξεργασία των κειμένων, ώστε η είσοδος να είναι σε μια πιο κατάλληλη μορφή για τους γράφους λέξεων. Για το σκοπό αυτό χρησιμοποιήθηκαν οι μέθοδοι *coreference resolution* και *collocation detection,* οι οποίες έχουν στόχο τη δημιουργία πιο αντιπροσωπευτικών ακμών και κόμβων αντίστοιχα. Στη συνέχεια, εξετάστηκε η χρήση *word embeddings* για την απόδοση βάρους στις ακμές των κόμβων, μέσω της απόστασης ομοιότητας των όρων. Για να αντιμετωπιστεί το πρόβλημα των γράφων λέξεων να δώσουν τα σωστό βάρος στους όρους που βρίσκονται στα δύο άκρα του κειμένου, αναπτύχθηκαν οι μέθοδοι ενίσχυσης των κόμβων, *Rebase* και *Boost*. Επιπλέον, η μέθοδος *μεταβλητού μήκους παραθύρου* επεκτείνει τις προηγούμενες και επιτρέπει σε κάθε όρο να έχει το δικό του μέγεθος παραθύρου. Με αυτόν τον τρόπο, μπορούμε να δώσουμε μεγαλύτερο βάρος σε κάποιο όρο ανάλογα με τη θέση του στο κείμενο και να διακρίνουμε ποια μέρη του κειμένου είναι σημαντικά για το πρόβλημα που αντιμετωπίζουμε και ποια όχι. Τέλος, εξετάστηκε και η χρήση ensembles γράφων λέξεων, ώστε να εκμεταλλευτούμε τις διάφορες επιλογές που υπάρχουν για την κατασκευή των γράφων. Η συνεισφορά των μεθόδων αναδεικνύεται από τα αποτελέσματα σε δύο διαφορετικά σύνολα δεδομένων, το 20 Newsgroups και το Reuters8.

1.3 Διάρθρωση της Εργασίας

Στο Κεφάλαιο 2 παρουσιάζεται το θεωρητικό υπόβαθρο της ταξινόμησης κειμένου με μεθόδους μηχανικής μάθησης και το μοντέλο γράφών λέξεων. Στη συνέχεια γίνεται αναφορά στις σχετικές εργασίες που χρησιμοποιούν τους γράφους λέξεων, ενώ περιγράφονται και οι μέθοδοι αξιολόγησης για την εργασία της ταξινόμησης.

Στο κεφάλαιο 3 εισάγονται διάφορες τροποποιήσεις στο μοντέλο γράφων λέξεων για την ταξινόμηση που προτείνονται στο πλαίσιο αυτής της εργασίας. Οι τροποποιήσεις αφορούν αλλαγές τόσο στην προεπεξεργασία κειμένου όσο και στην δομή και τον τρόπο κατασκευής των γράφων λέξεων. Επιπλέον, παρουσιάζονται τα κίνητρα που οδήγησαν σε αυτές τις τροποποιήσεις και οι αδυναμίες των γράφων λέξεων που εντοπίστηκαν.

Στο Κεφάλαιο 4 αναλύονται οι τεχνικές λεπτομέρειες υλοποίησης των προτεινόμενων μεθόδων και παρουσιάζονται οι συλλογές κειμένων που χρησιμοποιούνται για την αξιολόγηση.

Το Κεφάλαιο 5 περιέχει την αξιολόγηση των τροποποιήσεων, όπου κάθε τροποποίηση εφαρμόζεται ξεχωριστά και συγκρίνεται με το βασικό μοντέλο γράφων λέξεων για να καθοριστεί η αποτελεσματικότητά της.

Τέλος, στο Κεφάλαιο 6 παρουσιάζονται τα συμπεράσματα που προέκυψαν για τις μεθόδους από την πειραματική διαδικασία και προτείνονται πιθανές επεκτάσεις για τους γράφους λέξεων και τις μεθόδους που εξετάστηκαν, που ενδέχεται να φανούν χρήσιμες για την ταξινόμηση κειμένων με χρήση γράφων λέξεων.

Κεφάλαιο 2

Θεωρητικό Υπόβαθρο

Το κεφάλαιο αυτό παρουσιάζει το θεωρητικό υπόβαθρο για την ταξινόμηση κειμένων με χρήση γράφων λέξεων. Αρχικά, περιγράφεται η γενική διαδικασία για την αυτόματη ταξινόμηση κειμένων με μεθόδους μηχανικής μάθησης και το TF-IDF. Στη συνέχεια, παρουσιάζεται το μοντέλο γράφων λέξεων, η διανυσματική αναπαράσταση των κειμένων μέσω του TW-IDF και οι σχετικές εργασίες που χρησιμοποιούν αυτό το μοντέλο. Τέλος, γίνεται αναφορά στις μεθόδους αξιολόγησης που χρησιμοποιούνται για το πρόβλημα της ταξινόμησης κειμένου.

2.1 Ταξινόμηση κειμένου

Το πρόβλημα της ταξινόμησης κειμένου είναι ένα από τα πιο κλασικά προβλήματα στο τομέα της επεξεργασίας φυσικής γλώσσας και συνίσταται στη σωστή ταξινόμηση των κειμένων μιας συλλογής σε μια ή περισσότερες κλάσεις. Ενδιαφέρουσες εφαρμογές είναι η ανίχνευση ανεπιθύμητης αλληλογραφίας, η ταξινόμηση ειδήσεων, η οργάνωση αρχείων και η αναγνώριση απόψεων, όπου οι συλλογές των κειμένων κυμαίνονται από ειδησεογραφικά ή επιστημονικά άρθρα μέχρι μηνύματα μεταξύ χρηστών και περιγραφές προϊόντων. Η πιο απλή μορφή του προβλήματος είναι όταν όλα τα κείμενα ανήκουν στη ίδια κατηγορία η οποία μπορεί να περιέχει πολλές κλάσεις, οπότε και πρόκειται για ταξινόμηση μονής ετικέτας πολλαπλών τάξεων (single-label multiclass classifications).

Η συνήθης προσέγγιση για την επίλυση αυτού του προβλήματος από τη μηχανική μάθηση προϋποθέτει πρώτα από όλα την αναπαράσταση του κειμένου στο μοντέλο διανυσματικού χώρου, δηλαδή τη μετατροπή του κειμένου σε διάνυσμα μέσω κάποιας διαδικασίας εξαγωγής χαρακτηριστικών. Πιο τυπικά, έστω ότι $D = \{d_1, d_2, \dots, d_n\}$ είναι το σύνολο των κειμένων και $T = \{t_1, t_2, \dots, t_m\}$ το σύνολο των m όρων που περιέχονται στο D. Οι όροι ενός κειμένου δεν ταυτίζονται με τις λέξεις που το αποτελούν, αλλά προκύπτουν από αυτές, αφού προηγηθεί κάποια προεπεξεργασία στις λέξεις, όπως η εύρεση της ρίζας ή η αφαίρεση και αντικατάσταση συμβόλων. Σε κάθε κείμενο d θέλουμε να αντιστοιχήσουμε ένα διάνυσμα $[w_{t_1}, w_{t_2}, \dots, w_{t_m}]$ όπου το βάρος w_{t_i} αντιπροσωπεύει τη σημασία του όρου t_i στο κείμενο. Αν ο όρος δεν περιέχεται στο κείμενο αυτό, τότε το βάρος του είναι ίσο με μηδέν. Συνολικά λοιπόν, σχηματίζεται ένας πίνακας μεγέθους $n * m$ ο οποίος θα αποτελέσει την είσοδο στο μοντέλο μηχανικής μάθησης που θα χρησιμοποιηθεί. Οι γραμμές του πίνακα αντιστοιχούν στα κείμενα και οι στήλες στους όρους, οπότε κάθε στοιχείο (i, j) του πίνακα είναι το βάρος του όρου t_j στο κείμενο i. Τα διανύσματα που παράγονται από αυτή τη διαδικασία έχουν συχνά μεγάλη διάσταση, καθώς το πλήθος των όρων που εμφανίζονται σε όλα τα κείμενα είναι μεγάλο, αλλά ταυτόχρονα είναι και πολύ αραιά, αφού σε κάθε κείμενο εμφανίζεται συνήθως μόνο ένα μικρό υποσύνολο όλων των όρων.

Μετά από τη μετατροπή των κειμένων σε διανύσματα χρειάζεται να οριστεί ένα μοντέλο ταξινομητή (classifier) από το χώρο της μηχανικής μάθησης. Η εκπαίδευση του μοντέλου θα είναι με επίβλεψη (supervised learning), εφόσον οι κλάσεις στις οποίες ανήκουν τα κείμενα είναι γνωστές. Για την προπόνηση και την αξιολόγηση του ταξινομητή, το σύνολο δεδομένων χωρίζεται σε δύο υποσύνολα, το σύνολο εκπαίδευσης (train set) και το σύνολο αξιολόγησης (test set). Στη συνέχεια, το μοντέλο εκπαιδεύεται μέσω του πρώτου συνόλου και παράγει προβλέψεις για το δεύτερο, το οποίο δεν θα του έχει δοθεί ποτέ ως είσοδος. Τις προβλέψεις αυτές μπορούμε να τις συγκρίνουμε με την πραγματική κλάση στην οποία ανήκει κάθε κείμενο, για να αξιολογήσουμε πόσο καλά λειτουργεί το μοντέλο.

Για το πρόβλημα της ταξινόμησης κειμένου, οι γραμμικές μηχανές διανυσμάτων υποστήριξης (Linear Support Vector Machine ή Linear SVM) έχουν δώσει παραδοσιακά πολύ καλά αποτελέσματα, αφού μπορούν να διαχειριστούν πολύ καλά εισόδους όπου τα δεδομένα είναι πολύ αραιά και το πλήθος των διαστάσεων είναι πολύ μεγαλύτερο από το πλήθος των δειγμάτων.

2.2 Διανυσματική αναπαράσταση κειμένου

2.2.1 Bag of Word και TF-IDF

Δύο από τις πιο συνηθισμένες και παλιές μεθόδους αναπαράστασης κειμένου σε διάνυσμα είναι το bag-of-words και το TF-IDF [2], τα οποία χρησιμοποιούνται ακόμα και σήμερα [3] και αποτελούν ένα τυπικό σημείο αναφοράς και σύγκρισης για τα νέα μοντέλα.
Στο μοντέλο bag-of-words (BoW) κάθε κείμενο αναπαρίσταται από ένα ασκί (bag) που περιέχει όλους τους μοναδικούς όρους που εμφανίζονται σε αυτό και σε κάθε όρο αντιστοιχίζονται οι φορές εμφάνισης αυτού στο κείμενο:

$$BoW = \{(t_i, N_{t_i})\}, i = 1, \dots, n_d$$

όπου n_d το πλήθος των διαφορετικών όρων στο κείμενο d.

Η διανυσματική αναπαράσταση TF-IDF πρόκειται για επέκταση του bag-of-words. Αποτελείται από δύο βασικά κομμάτια, το TF (Term Frequencies) και το IDF (Inverted Document Frequencies) τα οποία και πολλαπλασιάζονται μεταξύ τους για να πάρουμε την τελική τιμή του βάρους ενός όρου σε ένα κείμενο. Η βασική εκδοχή του μοντέλου είναι:

$$TF - IDF(t, d, C) = tf(t, d) * idf(t, C)$$

$$TF - IDF(t, d, C) = \frac{BoW(t, d)}{n_D} * \log\left(\frac{|C| + 1}{|\{d: t \in d\}|}\right)$$

όπου με t συμβολίζεται ο όρος, με d το κείμενο, με C το σύνολο των κειμένων και με $|\cdot|$ η πληθικότητα ενός συνόλου.

Ο όρος αντίστροφης συχνότητας κειμένου (IDF) παρουσιάστηκε για πρώτη φορά στο [4] και είναι πολύ σημαντικός, καθώς μας επιτρέπει να μειώσουμε το βάρος σε κάποιο όρο ο οποίος εμφανίζεται σε πολλά κείμενα και άρα δεν μας παρέχει ικανοποιητική πληροφορία για το είδος του κειμένου. Αντίστροφα, μπορούμε να δώσουμε έμφαση σε κάποιον όρο ο οποίος εμφανίζεται σε λίγα κείμενα και είναι αρκετά ενδεικτικός για την κλάση τους.

2.2.2 Αδυναμίες του Bag of Words

Μια βασική παραδοχή που κάνει το bag of words είναι αυτή της ανεξαρτησίας των όρων, δηλαδή δεν λαμβάνει υπόψη του τη σειρά των όρων και κάθε όρος αντιμετωπίζεται με τον ίδιο τρόπο ανεξαρτήτως της θέσης του μέσα στο κείμενο, ενώ η φυσιολογική επέκταση του ώστε να χρησιμοποιεί n-grams δεν λύνει το πρόβλημα, καθώς και πάλι η πληροφορία για τη σχετική θέση των n-grams χάνεται. Το πλαίσιο μέσα στο οποίο εμφανίζεται μια λέξη και συγκεκριμένα γύρω από ποιες λέξεις βρίσκεται, είναι χαρακτηριστικό για την ίδια τη λέξη και για τη σημασιολογία της. Αν σε μια πρόταση αποκρύψουμε μια λέξη, ο άνθρωπος μπορεί συνήθως εύκολα να τη συμπληρώσει με την ίδια ή με μια συνώνυμη της, αφού από το περιεχόμενο και μόνο μπορεί να κρίνει ποιες λέξεις είναι κατάλληλες και ποιες όχι.

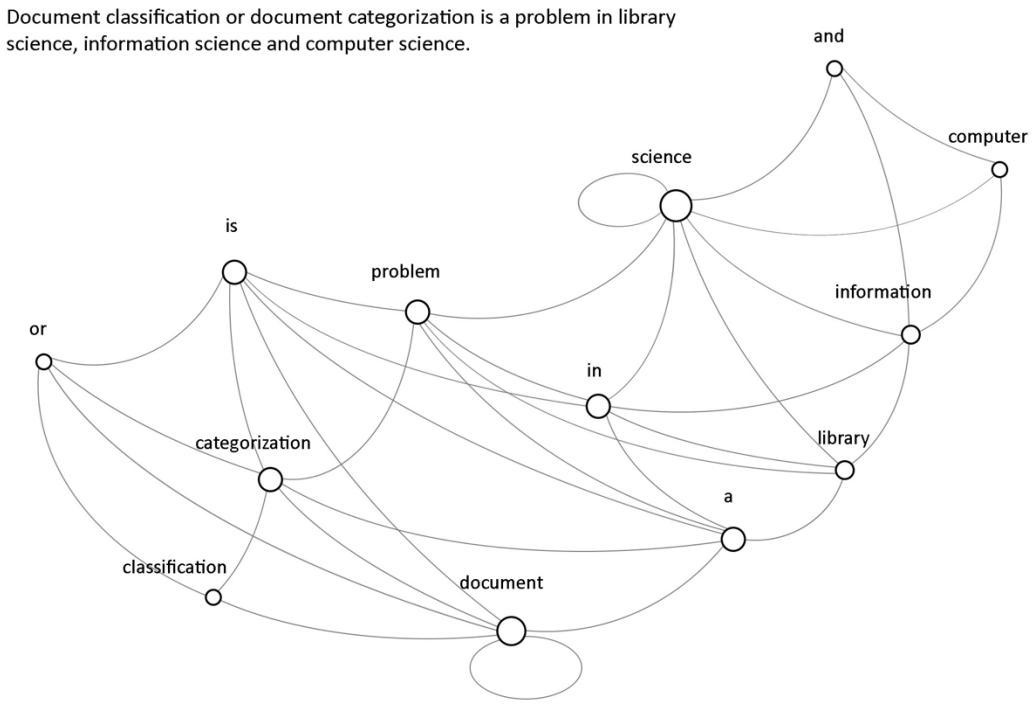
Ο γλωσσολόγος John Rupert Firth είχε αναφέρει χαρακτηριστικά: «You shall know a word by the company it keeps» [5, σελ 11] . Η συντροφιά (company) των λέξεων είναι φυσικά οι κοντινές της λέξεις μέσα στο κείμενο. Ως περιεχόμενο μιας λέξης μπορούμε τότε να ορίσουμε ένα παράθυρο που εκτείνεται μπροστά ή και πίσω απ' αυτήν και περιέχει όλες τις λέξεις που περικλείει. Το παράθυρο αυτό ορίζει ένα σύνολο λέξεων που σχετίζονται με κάποιο τρόπο ακριβώς επειδή βρίσκονται σε κοντινές θέσεις στο κείμενο και δεν χρειάζεται να είναι πολύ μεγάλο, αν θεωρήσουμε ότι η επιρροή μιας λέξης φθίνει σε σχέση με την απόσταση.

Την προσέγγιση αυτή έχουν ακολουθήσει τα μοντέλα Word2Vec [5] και GloVe [6] για να παράγουν διανύσματα λέξεων (word embeddings) τα οποία θα κωδικοποιούν μεταξύ άλλων τη σημασία μιας λέξης. Το GloVe χρησιμοποιεί ένα πίνακα συνύπαρξης (co-occurrence matrix) που περιέχει πληροφορίες για το πόσο συχνά εμφανίζεται μια λέξη στο περιεχόμενο μιας άλλης, δηλαδή μέσα στο ίδιο παράθυρο. Το Word2Vec και συγκεκριμένα η Skip-gram εκδοχή, χρησιμοποιεί και αυτή ένα κινούμενο παράθυρο πάνω στους όρους του κειμένου, αλλά με λίγο διαφορετικό τρόπο. Αντί να σχηματίζει ένα πίνακα συνύπαρξης, σχηματίζει ζεύγη εισόδου-εξόδου με βάση τους όρους που βρίσκονται μέσα στο ίδιο παράθυρο και προσπαθεί με βάση έναν κεντρικό όρο να προβλέψει τις κοντινές του λέξεις. Σε αυτή τη περίπτωση, το παράθυρο που καθορίζει το περιεχόμενο εκτείνεται και μπροστά και πίσω από τον κεντρικό όρο και οι λέξεις που βρίσκονται πιο κοντά στην κεντρική λέξη έχουν και μεγαλύτερο βάρος.

Το ενδιαφέρον είναι ότι και στις δύο περιπτώσεις η σημασία των όρων προκύπτει από τη συνύπαρξη τους σε ένα παράθυρο με βάση μόνο το περιεχόμενο τους και δεν χρησιμοποιείται καμιά εξωτερική γνώση για τον κόσμο ή για το αντικείμενο του κειμένου. Παρόμοια τεχνική χρησιμοποιούν και οι γράφοι λέξεων, όπου πάλι χρησιμοποιείται ένα κινούμενο παράθυρο για να σχημαιστεί ο γράφος που αναπαριστά το κείμενο.

2.3 Γράφοι λέξεων

Οι γράφοι αποτελούν μια φυσιολογική μορφή αναπαράστασης σχέσεων εξαρτήσεων μεταξύ διαφόρων οντοτήτων και συνεπώς είναι μια κατάλληλη δομή για την αναπαράσταση κειμένου. Οι κόμβοι μπορεί να είναι μια συλλαβή, ένας όρος, μια πρόταση ή ένα ολόκληρο κείμενο, οι ακμές μπορεί να καθοριστούν με βάση τη σύνταξη, τη σημασιολογία ή τη συνύπαρξη των κόμβων και ο συνολικός γράφος μπορεί να αντιστοιχεί σε ένα κείμενο ή σε πολλά. Το [7] περιέχει μια ανασκόπηση των διαφόρων τρόπων με τους οποίους οι γράφοι έχουν χρησιμοποιηθεί για την αναπαράσταση κειμένου. Η παρούσα διπλωματική μελετάει το μοντέλο γράφων λέξεων (Graph of Words) όπως ορίστηκε στο [1].



Εικόνα 2.1: Αναπαράσταση κειμενου από μη κατευθυνόμενο γράφο χωρίς βάρος για μέγεθος παραθύρου 4. Το μέγεθος των κόμβων είναι ανάλογο του βαθμού

Στο μοντέλο γράφων λέξεων το κείμενο αναπαρίσταται ως ένας γράφος του οποίου οι κόμβοι είναι οι ξεχωριστοί όροι του κειμένου και οι ακμές συμβολίζουν τη συνύπαρξη δύο όρων σε ένα κινούμενο παράθυρο το οποίο διατρέχει όλο το κείμενο. Η βασική υπόθεση που κάνει το μοντέλο γράφων λέξεων είναι ότι οι λέξεις μέσα σε ένα κείμενο σχετίζονται μεταξύ τους και το κινούμενο παράθυρο προσπαθεί να συλλάβει ακριβώς αυτή τη σύνδεση. Μια λέξη δεν εμφανίζεται μόνη της μέσα σε ένα κείμενο και για να την ερμηνεύσουμε κατάλληλα πρέπει να κοιτάξουμε και τις λέξεις που εμφανίζονται γύρω από αυτήν. Πιο γενικά, είναι λογικό να υποθέσουμε ότι η συσχέτιση δύο λέξεων είναι αντιστρόφως ανάλογη της απόστασης τους μέσα στο κείμενο. Επιπλέον η σειρά με την οποία εμφανίζονται οι λέξεις μπορεί να είναι σημαντική πληροφορία και αυτό λαμβάνεται υπόψη με την κατεύθυνση των ακμών. Το μοντέλο δεν επιχειρεί καμία ερμηνεία των όρων και αποτελεί ένα στατιστικό μοντέλο, αφού βασίζεται απλά και μόνο στην συνύπαρξη των όρων μέσα σε ένα παράθυρο. Με αυτόν τον τρόπο, οι γράφοι λέξεων προσπαθούν να δώσουν έμφαση στη σειρά με την οποία εμφανίζονται οι όροι μέσα στο κείμενο και στο περιεχόμενο των λέξεων, όπως αυτό ορίζεται με τη βοήθεια του παραθύρου.

Για την κατασκευή του γράφου έχουμε αρκετές επιλογές:

- Κατευθυνόμενος ή μη: Με τους κατευθυνόμενους γράφους μπορούμε να διατηρήσουμε τη φυσική ροή του κειμένου, ενώ στους μη κατευθυνόμενους η ακμή συμβολίζει απλά τη συνύπαρξη των όρων.

- Με βάρος ή χωρίς: Αν ένα ζευγάρι όρων συνυπάρχει πολλές φορές μέσα σε ένα παράθυρο ή και σε διαφορετικά, είναι μια πληροφορία που πιθανώς να είναι χρήσιμη. Στους γράφους με βάρος, το βάρος κάθε ακμής είναι ανάλογο του πόσες φορές θα συναντήσουμε αυτή τη ακμή, ενώ στους γράφους χωρίς βάρος δεν αναθέτουμε κάποιο βάρος στις ακμές.
- Μέγεθος κινούμενου παραθύρου: Το μέγεθος του παραθύρου έχει μεγάλη σημασία, γιατί αυτό καθορίζει στην ουσία ποιες ακμές θα δημιουργηθούν μέσα στο γράφο και ποιοι όροι θα συνδεθούν μεταξύ τους. Πρακτικά θα θέλαμε το παράθυρο να είναι αρκετά μεγάλο ώστε να λαμβάνει υπόψη πολλές συνυπάρξεις όρων, όμως από την άλλη θα θέλαμε να είναι σχετικά μικρό, γιατί ο χρόνος κατασκευής του γράφου εξαρτάται γραμμικά από το μέγεθος του παραθύρου. Ενδεικτικά, τα μεγέθη παραθύρων στη βιβλιογραφία κυμαίνονται μέχρις στιγμής από 2 μέχρι 10.

Το ζήτημα βέβαια δεν είναι να μείνουμε στην αναπαράσταση του κειμένου ως γράφο, αλλά να καταλήξουμε σε μια διανυσματική αναπαράστασή του. Με άλλα λόγια πρέπει να αναθέσουμε κάποιο βάρος σε κάθε κόμβο/όρο του γράφου που να μεταφράζει τη σημασία του κόμβου στη σημασία του όρου στο κείμενο. Οι Rousseau και Vazirgiannis στο [1] πρότειναν το βάρος του κάθε όρου να είναι ο βαθμός του αντίστοιχου κόμβου στο γράφο, καθώς πρόκειται για ένα απλό και εύκολο υπολογίσιμο μέτρο σε σύγκριση με άλλα που βασίζονται στη κεντρικότητα των κόμβων ή στα συντομότερα μονοπάτια και το οποίο δίνει και καλύτερα αποτελέσματα. Ο βαθμός ενός κόμβου είναι ανάλογος του πλήθους των γειτονικών κόμβων και άρα οι πιο σημαντικοί όροι θα είναι αυτοί που βρίσκονται σε πολλά παράθυρα και συνυπάρχουν με πολλές διαφορετικές λέξεις μέσα σε αυτά. Στη συνέχεια, με βάση το βάρος κάθε όρου προκύπτει το μοντέλο TW-IDF σε αναλογία με το πιθανοτικό TF-IDF, όπου η συχνότητα ενός όρου (term frequency) αντικαθίστανται από το βάρος του κάθε όρου (term weight).

Ειδικότερα, για να πάρουμε την τιμή TW-IDF ενός όρου, εφαρμόζουμε μια συνάρτηση κανονικοποίησης του μήκους του κειμένου (pivoted document length normalization) στο βαθμό κάθε κόμβου και στη συνέχεια πολλαπλασιάζουμε με το IDF όπως και στο TF-IDF μοντέλο:

$$TW - IDF(t, d, C) = \frac{tw(t, d)}{1 - b + b * \frac{|d|}{avdl}} * \log\left(\frac{|C| + 1}{|\{d: t \in d\}|}\right)$$

όπου b σταθερά ίση με 0.003 και $avdl$ το μέσο μήκος των κειμένων στη συλλογή C.

Αρχικά, οι γράφοι λέξεων εφαρμόστηκαν στο πρόβλημα της ανάκτησης πληροφοριών (Information Retrieval ή IR), αλλά στη συνέχεια χρησιμοποιήθηκαν και σε άλλα πεδία της επεξεργασίας φυσικής γλώσσας. Ένα από τα βασικά προτερήματα των γράφων λέξεων είναι ότι έχοντας αναπαραστήσει το κείμενο ως ένα γράφο, έχουμε αποκτήσει μια ενδιάμεση αναπαράσταση μεταξύ του κειμένου και του διανύσματος χαρακτηριστικών. Αυτό μας επιτρέπει να εφαρμόσουμε διάφορους αλγορίθμους γράφων για να επεξεργαστούμε, να εμπλουτίσουμε και να εξάγουμε πληροφορίες από αυτόν. Το Textrank [8] κατασκευάζει παρόμοιους γράφους με αυτούς του μοντέλου γράφων λέξεων και χρησιμοποιεί τον αλγόριθμο Pagerank, για να κατατάξει τους κόμβους/όρους και να πάρει τελικά τις λέξεις κλειδιά από ένα κείμενο. της Στα [9], [10], [11] χρησιμοποίησαν μεθόδους εκφυλισμού των γράφων, όπως το k-core και το k-truss, για να πάρουν συνεκτικούς υπογράφους τους

οποίους στη συνέχεια χρησιμοποιούν για εύρεση λέξεων κλειδιών (Keyword Extraction) και για σύνοψη κειμένου (Extractive Summarization). Στο [12] διερευνώνται διάφοροι μέθοδοι για την ανάθεση βάρους στους κόμβους για το πρόβλημα της ταξινόμησης κειμένου (Text Categorization) και στο [13] οι γράφοι λέξεων χρησιμοποιήθηκαν για το πρόβλημα της μοντελοποίησης θεμάτων (Topic Modelling).

Μια άλλη προσέγγιση είναι η χρήση άλλων μεθόδων εξαγωγής χαρακτηριστικών από τους γράφους πέρα από το TW-IDF, όπως στο [14] και στο [15] όπου τα προβλήματα της ταξινόμησης κειμένου (Text Categorization) και της ομοιότητας κειμένων (Document Similarity) μετατρέπονται στα αντίστοιχα προβλήματα ταξινόμησης και ομοιότητα γράφων.

Τέλος, το GoWVis [16] είναι μια διαδραστική ηλεκτρονική πλατφόρμα στην οποία μπορεί κανείς να εισάγει το δικό του κείμενο και να φτιάξει τους αντίστοιχους γράφους λέξεων με διάφορες επιλογές, τόσο για την προεπεξεργασία του κειμένου, όσο για τον ίδιο το γράφο. Παράλληλα, με χρήση των αλγορίθμων εκφυλισμού γράφων παρουσιάζονται οι λέξεις κλειδιά και μια σύνοψη του κειμένου, ενώ με τη χρήση αλγορίθμων για εύρεση κοινοτήτων σε γράφους είναι δυνατόν να διαχωριστούν οι διαφορετικές ενότητες του κειμένου και να παρουσιαστούν τα παραπάνω αποτελέσματα ξεχωριστά.

2.4 Αξιολόγηση

2.4.1 Μετρικές

Για την αξιολόγηση των συστημάτων μηχανικής μάθησης και εν προκειμένω για το πρόβλημα της ταξινόμησης κειμένου με επιβλεπόμενη μάθηση, είναι απαραίτητο να έχουμε κάποια κείμενα των οποίων η κλάση να είναι γνωστή. Οι κλάσεις αυτές μπορεί να είναι δύο ή περισσότερες. Για παράδειγμα η ανεπιθύμητη αλληλογραφία είναι ανεπιθύμητη ή όχι, ενώ εάν τα κείμενα που έχουμε να ταξινομήσουμε είναι ειδησεογραφικά πρέπει να επιλέξουμε ανάμεσα από πολλές κλάσεις όπως πολιτική, αθλητική ή πολιτιστική είδηση. Στη πρώτη περίπτωση μιλάμε για δυαδικά προβλήματα, όπου η μία κλάση αναφέρεται ως θετική (Positive) και η άλλη ως αρνητική (Negative). Στη δεύτερη περίπτωση που η ταξινόμηση γίνεται ανάμεσα σε πολλές κλάσεις, τότε συνήθως θεωρούμε τη σωστή κλάση ως Positive και όλες τις άλλες ως Negative. Με βάση τα παραπάνω, σχηματίζεται ο εξής πίνακας λαθών (confusion matrix) που περιλαμβάνει όλες τις δυνατές ταξινομήσεις:

		Actual Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Πίνακας 2.1: Confusion matrix

Περιγραφικά τα στοιχεία του πίνακα είναι:

- True positive (TP): Το κείμενο είναι positive και η πρόβλεψη είναι positive
- True Negative (TN): Το κείμενο είναι negative και η πρόβλεψη είναι negative
- False Positive (FP): Το κείμενο είναι negative και η πρόβλεψη είναι positive
- False Negative (FN): Το κείμενο είναι positive και η πρόβλεψη είναι negative

Οι πιο δημοφιλείς μετρικές για την απόδοση ενός μοντέλου ταξινόμησης είναι το accuracy και το F_1 score. Το accuracy μετράει την ακρίβεια του μοντέλου, δηλαδή πόσες φορές η προβλεπόμενη κλάση είναι ίδια με την πραγματική:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Το F_1 score ορίζεται ως ο αρμονικός μέσος δύο άλλων μετρικών, του precision και του recall. Το precision είναι το ποσοστό των σωστών αποτελεσμάτων, αν περιοριστούμε μόνο στις Positive προβλέψεις, δηλαδή απ' όλες τις positive προβλέψεις που έκανε το μοντέλο πόσες ήταν στην πραγματικότητα σε αυτή τη κλάση:

$$Precision = \frac{TP}{TP + FP}$$

Η μετρική αυτή χρησιμοποιείται περισσότερο στις περιπτώσεις που τα positive παραδείγματα είναι λίγα συγκριτικά με τα negative και το accuracy μπορεί να δώσει αποπροσανατολιστικά αποτελέσματα. Συμπληρωματικό του precision είναι το recall το οποίο μετράει πόσα από τα positive κείμενα αναγνώρισε το μοντέλο σωστά ως τέτοια:

$$Recall = \frac{TP}{TP + FN}$$

Αν ένα μοντέλο κάνει μόνο positive προβλέψεις τότε το FN θα είναι 0, ενώ αντίστοιχα αν κάνει μόνο negative προβλέψεις, τότε το FP θα είναι 0. Συμπερασματικά, για να έχουμε υψηλό και το precision και το recall χρειάζεται να βρούμε μια ισορροπία, καθώς αυτές οι δύο μετρικές στην ουσία αντιμάχονται η μία την άλλη. Επειδή λοιπόν είναι δύσκολο να συγκρίνουμε μεταξύ διαφορετικών μοντέλων που έχουν διαφορετικό precision και recall, το F_1 score συνδυάζει αυτές τις δύο μετρικές, ώστε να μπορούμε με ένα αριθμό να αξιολογήσουμε τα συστήματα:

$$F_1 = 2 * \frac{precision * recall}{precision + recall}$$

2.4.2 Micro και Macro Average statistics

Στην περίπτωση της multiclass ταξινόμησης, οι παραπάνω μετρικές ορίζονται για κάθε μία κλάση ξεχωριστά, θεωρώντας κάθε φορά αυτή την κλάση ως Positive και τις άλλες ως Negative. Υπάρχουν δύο προσεγγίσεις για τον υπολογισμό του μέσου όρου των μετρικών αυτών, ώστε τελικά να προκύψει μια συνολική αξιολόγηση του μοντέλου: η micro και η macro average. Στην πρώτη προσέγγιση οι επιμέρους προβλέψεις TP, TN, FP, FN προστίθενται όλες μαζί για να προκύψει η ζητούμενη μετρική, ενώ στο macro average η τελική μετρική προκύπτει ως ο μέσος όρος των επιμέρους μετρικών. Αν για παράδειγμα μας ενδιαφέρει το recall και έχουμε k διαφορετικές κλάσεις τότε:

- $Micro\ average\ Recall = \frac{TP_1 + TP_2 + \cdots + TP_k}{(TP_1 + TP_2 + \cdots + TP_k) + (FN_1 + FN_2 + \cdots + FN_k)}$

- $Macro\ average\ Recall = \frac{\frac{TP_1}{TP_1+FN_1}+\frac{TP_2}{TP_2+FN_2}+\cdots+\frac{TP_k}{TP_k+FN_k}}{k} = \frac{Recall_1+Recall_2+\cdots+Recall_k}{k}$

Οι δύο αυτές προσεγγίσεις μπορεί να διαφέρουν αρκετά, ειδικά στην περίπτωση όπου οι κλάσεις δεν είναι ισοκατανεμημένες. Αν συμβαίνει αυτό, τότε η micro average προσέγγιση είναι καταλληλότερη, καθώς η macro δεν λαμβάνει υπόψη την πληθικότητα των κλάσεων και συμπεριφέρεται σαν όλες οι κλάσεις να έχουν το ίδιο βάρος.

Κεφάλαιο 3

Τροποποιήσεις του μοντέλου γράφων λέξεων

Στο κεφάλαιο αυτό παρουσιάζονται οι τροποποιήσεις του μοντέλου γράφων λέξεων που προτείνονται στο πλαίσιο αυτής της εργασίας, οι οποίες έχουν ως στόχο την σωστότερη αναπαράσταση των κειμένων από τους γράφους λέξεων και τη βελτίωση της απόδοσης αυτών στο πρόβλημα της ταξινόμησης κειμένου. Οι τροποποιήσεις αφορούν τόσο την προεπεξεργασία των κειμένων όσο και την κατασκευή των γράφων. Επιπλέον, εξετάζεται και η χρήση ensembles στους γράφους λέξεων για περαιτέρω βελτίωση της απόδοσης. Οι τεχνικές λεπτομέρειες για την υλοποίηση των διαφόρων μεθόδων αναλύονται στο Κεφάλαιο 4.

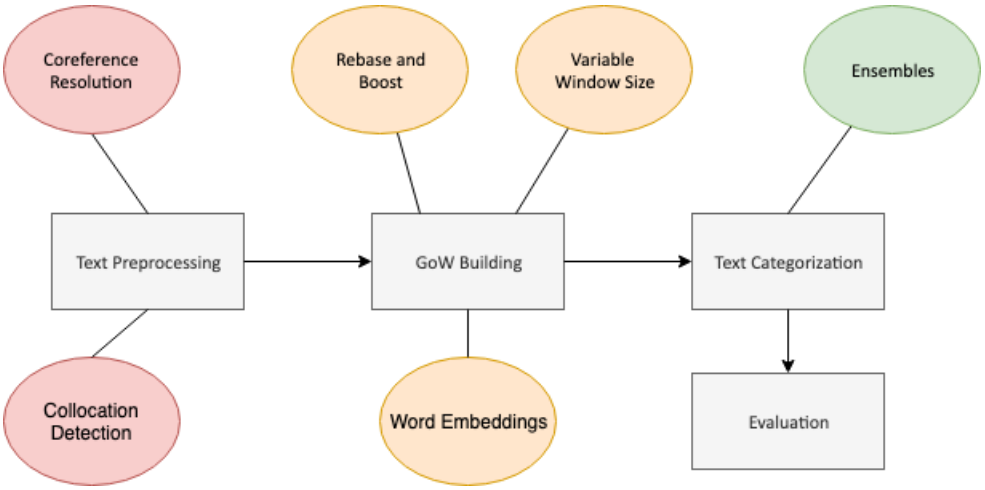
3.1 Εισαγωγή

Το μοντέλο γράφων λέξεων όπως παρουσιάστηκε στο Κεφάλαιο 2, αποτελεί ένα σχετικά νέο μοντέλο και παρόλο που έχει εφαρμοστεί σε πολλά προβλήματα επεξεργασίας φυσικής γλώσσας, στις σχετικές εργασίες ο βασικός πυρήνας του μοντέλου παραμένει πάντα ο ίδιος. Σε αυτές τις εργασίες, το μέγεθος παραθύρου είναι πάντα σταθερό και συνήθως μικρού μεγέθους (3 ή 4), δεν γίνεται κατάλληλη προεπεξεργασία του κειμένου και δεν δίνεται προσοχή στην εγγενή αδυναμία του μοντέλου να δώσει τα κατάλληλα βάρη στους όρους στην αρχή και στο τέλος του κειμένου. Οι τροποποιήσεις του μοντέλου που εξετάστηκαν σε αυτή την διπλωματική εργασία είναι:

- Προεπεξεργασία των κειμένων με χρήση coreference resolution και collocation detection
- Χρήση word embeddings για τα βάρη στις ακμές
- Ενίσχυση κόμβων μέσω των μεθόδων Rebase και Boost
- Μεταβλητό μέγεθος παραθύρου
- Ensembles γράφων λέξεων

Στο σχήμα 3.1 παρουσιάζεται η συνήθης διαδικασία που ακολουθείται για την ταξινόμηση κειμένων με χρήση των γράφων λέξεων μαζί με τις παραπάνω τροποποιήσεις.

Στη συνέχεια, αναλύονται περαιτέρω οι αδυναμίες των γράφων λέξεων που εντοπίστηκαν και οι αλλαγές που εξετάστηκαν.



Εικόνα 3.1: Διαδικασία ταξινόμησης με χρήση γράφων λέξεων

3.2 Προεπεξεργασία Κειμένου

Η προεπεξεργασία του κειμένου αποτελεί ένα πολύ βασικό κομμάτι του τομέα της επεξεργασίας φυσικής γλώσσας, καθώς είναι πολύ σημαντικό να μετασχηματιστεί το κείμενο σε μια μορφή κατάλληλη για το μοντέλο που χρησιμοποιείται κάθε φορά.

Πρώτα από όλα, χρειάζεται να γίνει καθαρισμός του κειμένου, όπου αυτό περιλαμβάνει συνήθως την αφαίρεση πολύ συχνών λέξεων (stopword removal), τη χρήση μικρών γραμμάτων, ώστε να υπάρχει ομοιομορφία στις λέξεις και τη διαγραφή των σημείων στίξης ή άλλων ανεπιθύμητων συμβόλων. Στη συνέχεια, το κείμενο πρέπει να κατατμηθεί στις βασικές μονάδες που το αποτελούν, τους όρους. Η εργασία αυτή ονομάζεται tokenization και διαχωρίζει τις συμβολοσειρές του κειμένου σε ατομικούς όρους, ανάλογα με τη γλώσσα και διάφορους κανόνες που ισχύουν για αυτή. Τέλος, υπάρχει η δυνατότητα να εφαρμοστεί και stemming ή lemmatization, ώστε οι όροι που θα προκύψουν να είναι πιο αντιπροσωπευτικοί. Ο στόχος και των δύο μεθόδων είναι να ανάγουν τη κάθε λέξη σε μια πιο βασική και κοινή μορφή, ώστε λέξεις όπως για παράδειγμα *look*, *looking* και *looks* να αναχθούν όλες στη λέξη *look*. Το stemming βασίζεται σε ευριστικές μεθόδους, για να κόψει μια λέξη στη ρίζα της ή καλύτερα στο κορμό της και υπάρχει περίπτωση ο παραγόμενος όρος να μην είναι πραγματική λέξη. Το lemmatization από την άλλη αντικαθιστά τη λέξη με το λήμμα της και είναι μια πιο σύνθετη διαδικασία, αφού απαιτεί πρώτα να έχει αναγνωριστεί ποιο μέρος του λόγου είναι κάθε λέξη (Part of Speech tagging).

Την παραπάνω διαδικασία ακολουθούν και οι περισσότερες σχετικές εργασίες που χρησιμοποιούν τους γράφους λέξεων. Ωστόσο, αν και η διαδικασία αυτή είναι τυπική, δεν λαμβάνει υπόψη της ότι οι όροι θα χρησιμοποιηθούν ως κόμβοι σε ένα γράφο, οι οποίοι θα συνδέονται με βάση την απόσταση τους μέσα στο κείμενο. Ο στόχος είναι ο γράφος λέξεων να αποτελεί μια αντιπροσωπευτική αναπαράσταση του κειμένου και στο πλαίσιο αυτό θα πρέπει να τροποποιήσουμε και να ενισχύσουμε την προεπεξεργασία των κειμένων με πιο εξειδικευμένες τεχνικές, κατάλληλες για τους γράφους λέξεων.

Η καταλληλότητα θα πρέπει να αφορά και τους κόμβους και τις ακμές του γράφου. Οι κόμβοι του γράφου θα πρέπει να είναι όσο πιο αντιπροσωπευτικοί γίνεται των οντοτήτων που εμφανίζονται μέσα στο κείμενο και δεν χρειάζεται να αντιστοιχούν μόνο σε ένα όρο του κειμένου. Παράλληλα, η σύνδεση των κόμβων μέσω ακμών μπορεί να βασίζεται όχι μόνο

στην συνύπαρξη των όρων, αλλά και στη ερμηνεία των όρων μέσα στο κείμενο. Για να γίνουν πιο κατανοητά τα παραπάνω, ας θεωρήσουμε την εξής πρόταση:

Barack Hussein Obama II is an American politician, born on 1961. He was the president of the United States from 2009 to 2017.

Όταν ένας άνθρωπος διαβάζει αυτή την πρόταση μπορεί να καταλάβει ότι οι όροι *Barack Hussein Obama II* αποτελούν μια οντότητα και αντιστοιχούν σε ένα φυσικό πρόσωπο και συνεπώς μπορεί να συμπεράνει ότι δεν πρέπει να θεωρήσει τους όρους αυτούς ανεξάρτητα τον ένα από τον άλλον, αλλά να τους αντιμετωπίσει ως ένα σύνολο. Επιπλέον, μπορεί να καταλάβει ότι η δεύτερη πρόταση αναφέρεται ουσιαστικά στην οντότητα *Barack Hussein Obama II* και η αντωνυμία *He* χρησιμοποιείται απλώς για αναφορά και δεν παίζει κάποιο άλλο ρόλο. Ωστόσο, στον αντίστοιχο γράφο λέξεων που έχει σχηματιστεί με μικρό μέγεθος παραθύρου και χωρίς κάποια άλλη προεπεξεργασία πέρα του tokenization, η οντότητα του Obama αντιπροσωπεύεται από τέσσερις διαφορετικούς κόμβους, ενώ δεν απεικονίζεται το γεγονός ότι ο Barack Obama ήταν πρόεδρος της Αμερικής, αφού δεν υπάρχει ακμή από τον κόμβο Obama προς του κόμβους president, United και State.

Για να αντιμετωπιστεί αυτό το πρόβλημα εξετάστηκαν δύο διαφορετικές τεχνικές προεπεξεργασίας, το coreference resolution και το collocation detection, που προηγούνται των συνηθισμένων και μπορούν να βοηθήσουν στο σχηματισμό καταλληλότερων ακμών και κόμβων αντίστοιχα.

3.2.1 Coreference Resolution

Το coreference resolution αφορά τον πλήρη καθορισμό των αναφορών μεταξύ των όρων του κειμένου. Η γνώση των αναφορών μπορεί να μας βοηθήσει να κατανοήσουμε ένα κείμενο καλύτερα και έχει αρκετές εφαρμογές σε διάφορα προβλήματα της επεξεργασίας φυσικής γλώσσας, όπως στη σύνοψη κειμένου και στην ανάκτηση πληροφοριών. Από την άλλη πλευρά, η διαδικασία εξαγωγής αναφορών είναι αρκετά δύσκολη και πολύπλοκη, καθώς οι τύποι των αναφορών μπορεί να είναι διαφόρων ειδών όπως anaphora, cataphora, exophora και noun coreferences, ανάλογα με το αν η αναφορά προηγείται της οντότητας ή αν η οντότητα αναφέρεται ρητά ή εννοείται.

Χαρακτηριστικό της δυσκολίας του προβλήματος είναι το Winograd Schema Challenge [17] το οποίο αποτελεί μια νέα πρόκληση για τα συστήματα τεχνητής νοημοσύνης, αντίστοιχο του γνωστού Turing test. Στο πρόβλημα αυτό, παρουσιάζονται δύο προτάσεις οι οποίες αν και διαφέρουν μόνο σε μια λέξη, το νόημα τους είναι πολύ διαφορετικό. Η πρόκληση αυτή αποτελείται από ένα τεστ ερωτήσεων πολλαπλών επιλογών, όπου η κάθε ερώτηση που τίθεται στο σύστημα τεχνητής νοημοσύνης αφορά το σε ποιά οντότητα της πρότασης αναφέρεται μια λέξη. Έστω ότι έχουμε τις εξής δύο προτάσεις:

Οι δημοτικοί σύμβουλοι δεν έδωσαν στους διαδηλωτές άδεια, γιατί αυτοί *φοβόντουσαν* τις βιοπραγίες.
Οι δημοτικοί σύμβουλοι δεν έδωσαν στους διαδηλωτές άδεια, γιατί αυτοί *υποστήριζαν* τις βιοπραγίες.

Στη πρώτη πρόταση, οι δημοτικοί σύμβουλοι φοβόντουσαν τις βιοπραγίες, ενώ στη δεύτερη οι διαδηλωτές υποστήριζαν τις βιοπραγίες. Το νοήμα των δύο προτάσεων είναι

πολύ διαφορετικό και για να ερμηνεύσει κανείς κατάλληλα την πρόταση πρέπει να ξέρει που αναφέρεται η λέξη *αυτοί*. Με άλλα λόγια, η εξαγωγή των αναφορών που περιέχει μια πρόταση προτείνεται στο Winograd Schema Challenge ως ένα μέτρο καθορισμού του κατά πόσο μια μηχανή μπορεί να σκεφτεί και να επιδείξει νοημοσύνη, αφού απαιτεί τη χρήση λογικής και γενικής γνώσης για το εξωτερικό κόσμο.

Οι παραδοσιακές προσεγγίσεις για το coreference resolution ήταν βασισμένες σε συστήματα κανόνων ενώ πιο πρόσφατα με την εμφάνιση της βαθιάς μηχανικής μάθησης έχουν γίνει σημαντικά βήματα αλλά ακόμα τα αποτελέσματα είναι συνήθως περιορισμένα σε απλές μορφές αναφορών.

Όσο αφορά τους γράφους λέξεων, η αξιοποίηση των αναφορών μπορεί να φανεί πολύ χρήσιμη. Όπως φάνηκε και στο παράδειγμα της προηγούμενης ενότητας, αν και η αντωνυμία *He* αναφέρεται στην οντότητα Barack Obama, αν χρησιμοποιηθεί ένα μικρό μήκος παραθύρου, ο όρος president δεν θα συνδεθεί με τους όρους Barack και Obama. Επιπλέον, η αναφορά μεταξύ όρων μπορεί να απέχει πολλές λέξεις ή προτάσεις μέσα στο κείμενο, όποτε δεν αποτελεί λύση η χρήση ενός μεγαλύτερου παράθυρου, ώστε να προκύψουν αυτές οι συνδέσεις φυσικά. Παρόλο δηλαδή που ένας όρος αναφέρεται σε μια οντότητα, η πληροφορία αυτή δεν θα περάσει μέσα στο γράφο, αφού για τη σύνδεση των όρων βασιζόμαστε μόνο στη συνύπαρξη και δεν λαμβάνουμε υπόψη τη σημασιολογία και την ερμηνεία των όρων.

Ο τρόπος με τον οποίο χρησιμοποιήθηκε το coreference resolution στην εργασία αυτή είναι απλός και βασίζεται στην αντικατάσταση μέσα στο κείμενο της αναφοράς από την οντότητα στην οποία αναφέρεται. Με την αντικατάσταση αυτή, η οντότητα θα εμφανίζεται και σε άλλα μέρη του κειμένου και θα μπορεί να σχηματίσει ακμές και με άλλους κόμβους/όρους πέρα από αυτούς της αρχικής γειτονιάς της. Εάν οι αναφερόμενες οντότητες είναι σημαντικές για το κείμενο, τότε οι νέες συνδέσεις που θα αποκτήσουν θα βοηθήσουν, ώστε οι όροι να αποκτήσουν μεγαλύτερο βάρος και ταυτόχρονα οι αντίστοιχοι κόμβοι θα γίνουν πιο κεντρικοί μέσα στο γράφο. Ως αποτέλεσμα, ο γράφος θα γίνει πιο συνεκτικός κάτι που μπορεί να φανεί χρήσιμο για τις περιπτώσεις που μας ενδοαφέρει η δομή του γράφου και οι συνεκτικοί υπογράφοι που σχηματίζονται.

3.2.2 Collocation Detection

Οι οντότητες που περιέχει ένα κείμενο είναι πολύ σημαντικές, καθώς είναι αρκετά χαρακτηριστικές για το είδος του κείμενου. Μια είδηση που αναφέρει αθλητές είναι κατά πάσα περίπτωση αθλητική, ενώ ένα κείμενο που περιέχει ονομασίες χωρών και πόλεων θα έχει πιθανότατα σχέση με τη γεωγραφία. Συνήθως όμως, οι οντότητες αυτές περιγράφονται μέσω φράσεων που αποτελούνται από δύο ή περισσότερες λέξεις, όπως οι οντότητες Barack Obama και United States στο παράδειγμα που αναφέρθηκε παραπάνω. Αν όμως για κάθε όρο της φράσης αντιστοιχηθεί και ένας ξεχωριστό κόμβο στο γράφο λέξεων, τότε αφενός θα υπάρχουν ακμές μεταξύ των όρων αυτών που δεν είναι απαραίτητα χρήσιμες, αφετέρου οι γειτονικοί όροι θα συνδεθούν με όλους τους επιμέρους όρους και όχι με έναν αντιπροσωπευτικό. Σε έναν σημασιολογικό γράφο, οι κόμβοι περιέχουν οντότητες και οι ακμές συμβολίζουν ιδιότητες ή χαρακτηριστικά του κόμβου. Έτσι και στους γράφους λέξεων, φαίνεται φυσιολογικό να είναι προτιμότερο οι οντότητες να είναι ένας κόμβος στο γράφο, για να υπάρχει μια πιο σωστή σύνδεση με τους γειτονικούς κόμβους. Δεν έχει νόημα στις περισσότερες περιπτώσεις να θεωρηθούν οι όροι ξεχωριστά, αφού στην πραγματικότητα

αποτελούν ένα σύνολο και αποκτούν σημασία ακριβώς μέσα στο σύνολο αυτό, ενώ έξω από τη φράση οι όροι αυτοί μπορεί να πάρουν διαφορετική ερμηνεία.

Γενικότερα, μέσα στα κείμενα εμφανίζονται ακολουθίες λέξεων οι οποίες συνυπάρχουν μαζί πιο συχνά από το αναμενόμενο. Τέτοιες ακολουθίες ονομάζονται collocations και μπορεί να περιγράφουν οντότητες όπως ανθρώπους ή οργανισμούς ή μπορεί απλά να είναι κάποιοι εξιδικευμένοι τεχνικοί όροι και ιδιωματισμοί. Η εξαγωγή collocation είναι πιο απλή από την αναγνώριση οντοτήτων, καθώς μπορεί να εφαρμοστεί σε οποιαδήποτε συλλογή κειμένων και δεν χρειάζεται κάποια γνώση για το κόσμο ή το είδος των οντοτήτων, αφού βασίζεται σε στατιστικές μεθόδους πάνω στα κείμενα της συλλογής. Αντίθετα, οι σύγχρονες μέθοδοι αναγνώρισης οντοτήτων χρησιμοποιούν τεχνητά νευρωνικά δίκτυα τα οποία έχουν προπονηθεί σε κάποιου είδους κείμενα και είναι ακατάλληλα, αν οι οντότητες που εμφανίζονται στα νέα κείμενα είναι διαφορετικές από αυτές στις οποίες έχει προπονηθεί.

Μετά την αναγνώριση των collocations από μια συλλογή κειμένων, οι όροι της έκφρασης μπορούν να ενωθούν σε ένα καινούργιο όρο, ώστε πλέον το collocation να αντιστοιχεί σε ένα κόμβο μέσα στο γράφο. Για παράδειγμα, με αυτό τον τρόπο, τα collocations *San Francisco* και *Golden State Warriors* θα μετατραπούν σε *San_Francisco* και *Golden_State_Warriors* και θα αντιπροσωπεύονται καλύτερα μέσα στο γράφο.

3.3 Word Embeddings

Όπως έχει ήδη αναφερθεί, η σύνδεση δύο κόμβων στους γράφους λέξεων βασίζεται απλά στη συνύπαρξη των δύο αντίστοιχων όρων μέσα σε ένα παράθυρο, ενώ και το βάρος της ακμής, στην περίπτωσή των γράφων με βάρη, βασίζεται μόνο στην πολλαπλότητα της εμφάνισης της ακμής. Στους γράφους με βάρη, το βάρος κάθε νέα ακμής είναι ίσο με ένα και αν κάποια ακμή επαναλαμβάνεται το βάρος της αυξάνεται κατά ένα, ενώ στους γράφους χωρίς βάρος οι επαναλαμβανόμενες ακμές αγνοούνται. Με άλλα λόγια, η προσέγγιση αυτή είναι καθαρά στατιστική, δεν ερμηνεύει τους ίδιους τους όρους του κειμένου και τους αντιμετωπίζει ως μια απλή ακολουθία αντικειμένων. Από τη στιγμή όμως που αντικείμενο των γράφων λέξεων είναι ακριβώς οι λέξεις, μπορούμε να προσαρμόσουμε τη διαδικασία κατασκευής των γράφων, ώστε να να χρησιμοποιούνται τα word embeddings των λέξεων.

Τα word embeddings είναι μια αναπαράσταση των λέξεων σε ένα διάνυσμα σχετικά μικρής διάστασης. Η ανάγκη για αυτό προκύπτει απ' το γεγονός ότι δεν γίνεται για παράδειγμα να συγκριθούν αμέσως δύο λέξεις, γιατί αναπαρίστανται συνήθως ως διακριτά αντικείμενα με ένα αναγνωριστικό, το οποίο δεν αντιπροσωπεύει κάτι συγκεκριμένο για τις λέξεις. Επιπλέον, η διακριτή μορφή των λέξεων δεν είναι κατάλληλη για τα περισσότερα μοντέλα μηχανικής μάθησης τα οποία αναμένουν στην είσοδο τους πραγματικά διανύσματα. Με τη χρήση λοιπόν των word embeddings, κάθε λέξη του κειμένου αντιστοιχίζεται σε ένα διάνυσμα πραγματικών αριθμών n διαστάσεων και ο στόχος είναι το διάνυσμα αυτό να αντιπροσωπεύει τη σημασιολογία και το περιεχόμενο γύρω από το οποίο εμφανίζεται ο όρος αυτός. Η αναπαράσταση με διανύσματα επιτρέπει μεταξύ άλλων την εκτέλεση διαφόρων μαθηματικών πράξεων μεταξύ των αντίστοιχων λέξεων. Αν δύο λέξεις είναι κοντά στο n-διάστατο χώρο, τότε θα εμφανίζονται συχνά γύρω από το ίδιο περιεχόμενο και επομένως θα έχουν παρόμοια σημασία.

Η πρώτη αρκετά πετυχημένη μέθοδος για τη παραγωγή των word embeddings με χρήση τεχνητών νευωνικών δικτύων είναι το Word2Vec το 2013 και έκτοτε έχει ακολουθήσει το GloVe και διάφορες τροποποιήσεις αυτών. Ένα χρήσιμο στοιχείο των μοντέλων αυτών είναι ότι μπορούν να εκπαιδευτούν πάνω σε πολύ μεγάλες συλλογές κειμένων, αφού δεν

χρειάζεται κάποια γνώση για το είδος των κείμενων και υπάρχουν διαθέσιμα αρκετά word embeddings τα οποία έχουν εκπαιδευτεί πάνω σε πολύ μεγάλα σύνολα δεδομένων. Εναλλακτικά, μπορεί κανείς να προπονήσει από την αρχή τα word embeddings σε οποιαδήποτε συλλογή κειμένων, σε περίπτωση που αυτή είναι αρκετά διαφορετική και ιδιαίτερη.

Ένας από τους τρόπους που μπορούμε να εισάγουμε τα word embeddings στην κατασκευή των γράφων είναι να χρησιμοποιήσουμε την απόσταση ομοιότητας των όρων για να καθορίσουμε το βάρος μιας ακμής. Οι ακμές θα προκύπτουν και πάλι μέσω ενός κινούμενου παραθύρου, αλλά το βάρος θα είναι συνάρτηση της ομοιότητας των όρων και οι όροι που έχουν ομοιότητα με πολλούς άλλους όρους θα έχουν μεγαλύτερο βαθμό και άρα μεγαλύτερη σημασία. Επιπλέον, μπορούμε να συνδυάσουμε τη σημασιολογική πληροφορία της απόστασης με το κλασικό βάρος των γράφων λέξεων.

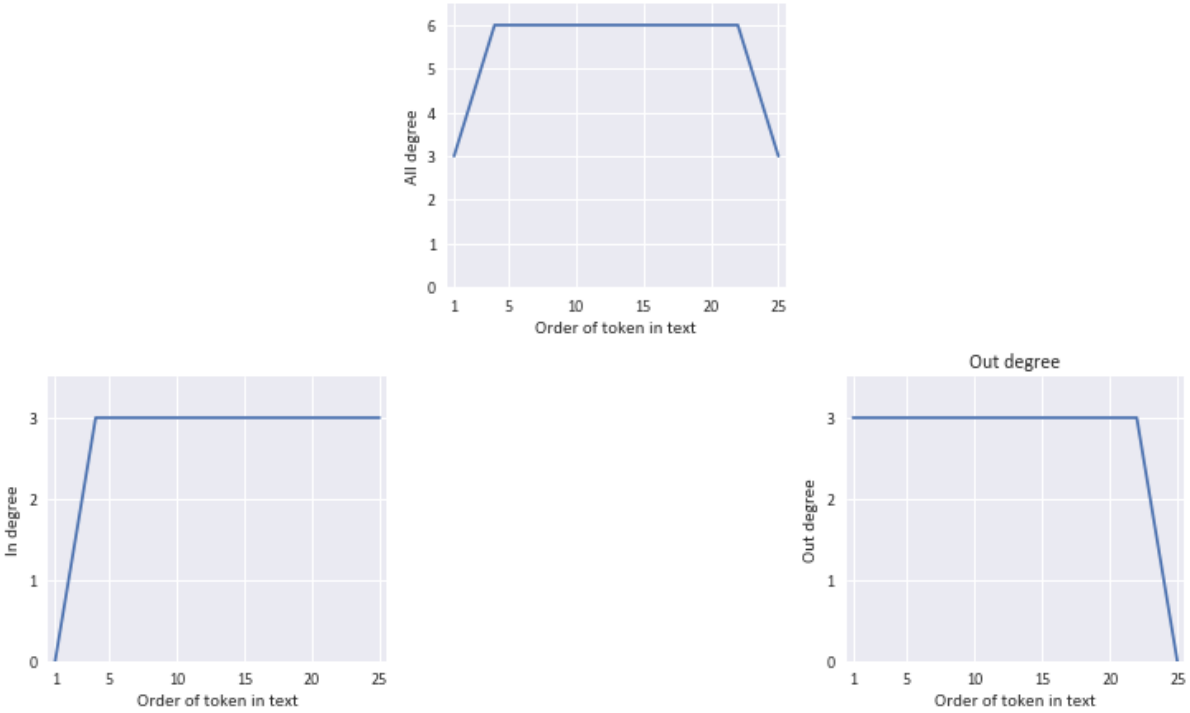
3.4 Ενίσχυση Κόμβων

Στο μοντέλο γράφων λέξεων το βάρος κάθε όρου προκύπτει από το βαθμό (στο εξής degree) που έχει ο αντίστοιχος κόμβος στον γράφο. Το degree ενός κόμβου στους μη κατευθυνόμενους γράφους είναι το πλήθος των γειτονικών κόμβων, ενώ οι βρόχοι (self loops) μετράνε για δύο. Στους κατευθυνόμενους γράφους, το degree είναι ίσο με το άθροισμα του in-degree και του out-degree. Το πρώτο είναι το πλήθος των ακμών που ξεκινάει από τον κόμβο και το δεύτερο το πλήθος των ακμών που καταλήγουν σε αυτόν. Το μέτρο αυτό είναι απλό, υπολογίζεται γρήγορα και αποδίδει καλύτερα στην ταξινόμηση σε σχέση με μετρικές που βασίζονται στην κλειστότητα ή στην κεντρικότητα του κόμβου, όπως προκύπτει από το [12].

Ωστόσο, το degree έχει ένα θεμελιώδες πρόβλημα στον τρόπο που αποδίδει βάρος στη αρχή και στο τέλος του κειμένου. Ας υποθέσουμε ότι έχουμε ένα κείμενο στο οποίο κάθε όρος εμφανίζεται μόνο μια φορά, ο αντίστοιχος γράφος είναι κατευθυνόμενος και έχει σχηματιστεί με μέγεθος παραθύρου ws το οποίο είναι αρκετά μικρότερο του μήκους του κειμένου. Σε αυτήν την περίπτωση, ο πρώτος όρος θα πρόκειται αναγκαστικά για πηγή, αφού κανένας κόμβος δεν μπορεί να δείχνει σε αυτόν, ενώ αντίστοιχα ο τελευταίος θα είναι καταβόθρα, επειδή δεν μπορεί να έχει εξερχόμενες ακμές. Παρόμοιο πρόβλημα έχουν όλοι οι κόμβοι στην αρχή και στο τέλος του κειμένου σε εύρος ws, αφού θα έχουν διαφορετικό degree από τους υπόλοιπους όρους. Πιο συγκεκριμένα, οι ανεπηρεάστοι κόμβοι που βρίσκονται στη μέση του κειμένου έχουν in-degree = out-degree = $ws - 1$. Κάθε όρος εμφανίζεται μια μόνο φορά στο κείμενο, οπότε είναι λογικό να δώσουμε το ίδιο βάρος σε κάθε όρο, αφού δεν έχουμε κάποιο τρόπο να τους διαχωρίσουμε. Το ίδιο συμβαίνει και στην αναπαράσταση TF-IDF, αφού όλοι οι όροι θα έχουν την ίδια συχνότητα και άρα το ίδιο βάρος.

Ωστόσο, στους γράφους λέξεων οι πρώτοι και οι τελευταίοι $ws - 1$ όροι θα έχουν μικρότερο βάρος από τους υπόλοιπους, απλά και μόνο γιατί βρίσκονται στα δύο άκρα του κειμένου. Αν χρησιμοποιήσουμε το in-degree ή το out-degree, τότε η ανισορροπία θα βρίσκεται στην αρχή ή στο τέλος του κειμένου αντίστοιχα, ενώ αν χρησιμοποιήσουμε το συνολικό degree, οι όροι που βρίσκονται και στην αρχή και στο τέλος δεν θα έχουν τα σωστά βάρη. Το πρόβλημα δεν περιορίζεται φυσικά μόνο στους κατευθυνόμενους γράφους, αφού το ίδιο συμβαίνει και στους μη κατευθυνόμενους. Πρόκειται για ένα μειονέκτημα που έχει το μοντέλο από τον τρόπο που ορίζεται και οφείλεται στο ότι τα κείμενα κάπου πρέπει να αρχίζουν και κάπου να τελειώνουν.

Στο διάγραμμα που ακολουθεί απεικονίζεται το φαινόμενο αυτό, όπου παρουσιάζεται το in-degree, το out-degree και το all-degree κάθε κόμβου σε συνάρτηση με τη θέση που έχει ο αντίστοιχος όρος στο ιδεατό κείμενο που περιγράφηκε:



Εικόνα 3.2: Άνιση αντιπροσώπευση όρων

Το φαινόμενο είναι πιο έντονο όσο πιο μεγάλο είναι το μέγεθος του παραθύρου, καθώς τόσο λιγότερες ακμές θα έχουν τα δύο άκρα του κειμένου. Βέβαια, σε ένα κανονικό κείμενο οι όροι που εμφανίζονται στα άκρα του κειμένου μπορεί να εμφανίζονται ξανά και σε άλλα σημεία του κειμένου, έτσι ώστε το βάρος τους να μην είναι τόσο μικρό. Παρόλα αυτά, το φαινόμενο θα συνεχίζει να υπάρχει ακόμα για τους μοναδικούς όρους, ενώ ανεξάρτητα από το εάν ένας όρος επαναλαμβάνετα, το βάρος που κερδίζει από την παρουσία του στην αρχή του κειμένου δεν είναι αντιπροσωπευτικό. Για να αντιμετωπιστεί αυτό το πρόβλημα εξετάστηκαν δύο προσεγγίσεις:

1. να τεθεί ένα τεχνητό όριο κάτω από το οποίο να μην επιτρέπεται να είναι το βάρος ενός κόμβου.
2. να αυξηθεί τεχνητά το βάρος των όρων στις άκρες του κειμένου, ώστε να αναπληρωθούν οι ακμές που τους λείπουν.

Στο εξής, η πρώτη μέθοδος θα αναφέρεται ως *Rebase* και η δεύτερη ως *Boost*. Οι δύο προσεγγίσεις διαφέρουν στο εύρος που επηρεάζουν τους κόμβους/όρους. Το Rebase επηρεάζει όλους τους κόμβους που έχουν degree μικρότερο του ορίου, οπότε και δεν περιορίζεται μόνο στους προβληματικούς όρους στις άκρες του κειμένου. Το όριο μπορεί να καθοριστεί με βάση τον τύπο του γράφου και με βάση τη μετρική που χρησιμοποιείται για το βάρος του κόμβου. Αντίθετα, η μέθοδος Boost αφορά αποκλειστικά τους όρους που βρίσκονται στις άκρες των κειμένων και μπορεί να ενισχύσει και κάποιο όρο που επαναλαμβάνεται και αργότερα μέσα στο κείμενο. Σε αυτήν την περίπτωση, το βάρος του πρώτου όρου θα αυξηθεί κατά window size-1, του δεύτερου window size-2, του τρίτου window size-3 κ.ο.κ, ενώ αντίστοιχες αυξήσεις θα πρέπει να γίνουν και στους τελευταίους

όρους. Περαιτέρω λεπτομέρειες για την υλοποίηση των μεθόδων περιέχονται στο Κεφάλαιο 4.

Το μειονέκτημα και με τις δύο μεθόδους είναι ότι αυξάνουν τεχνητά τα βάρη των όρων, χωρίς η αλλαγή αυτή να γίνεται αισθητή στον ίδιο το γράφο, αφού δεν εισάγουν καινούργιες ακμές. Επομένως, αυτές οι αλλαγές βοηθούν στην περίπτωση που ο γράφος χρησιμοποιείται μόνο για να σχηματιστεί ο TW-IDF πίνακας μέσω του βαθμού των κόμβων. Στην περίπτωση που αντικείμενο μελέτης είναι οι συνεκτικοί υπογράφοι των γράφων λέξεων, οι κόμβοι που αντιστοιχούν σε όρους στις δύο άκρες θα είναι και πάλι δύσκολο να συμμετάσχουν σε αυτούς, αφού έχουν λιγότερες ακμές από ότι θα έπρεπε.

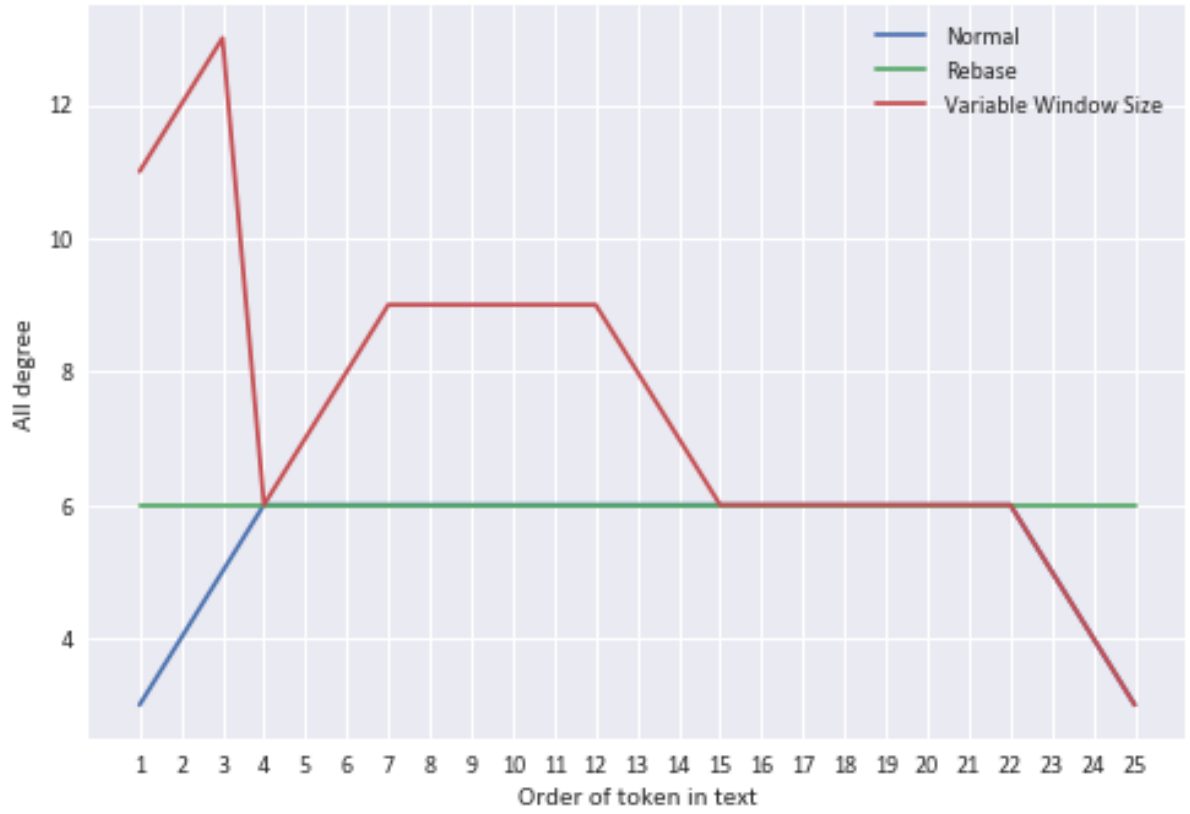
3.5 Μεταβλητό Μέγεθος Παραθύρου

Το πρόβλημα των γράφων λέξεων που παρατηρήθηκε στην προηγούμενη ενότητα είναι ότι δεν αντιπροσωπεύει σωστά τις δύο άκρες του κειμένου, γιατί οι κόμβοι σε αυτά τα σημεία έχουν λιγότερες ακμές. Ένας εναλλακτικός τρόπος για να αυξηθεί η σημασία του κάθε κόμβου/όρου μέσα στο γράφο είναι να χρησιμοποιηθεί μεγαλύτερο κινούμενο παράθυρο στους όρους αυτούς, ώστε να σχηματίζουν περισσότερες συνδέσεις. Με αυτόν τον τρόπο, αν ο πρώτος όρος έχει διπλάσιο μέγεθος παραθύρο, τότε δεν θα χρειάζεται να προστεθούν νοητές ακμές σε αυτόν, αφού θα έχει πραγματικές ακμές με περισσότερους όρους.

Στο παραδοσιακό μοντέλο γράφων λέξεων, το κινούμενο παράθυρο έχει σταθερό μήκος για όλους τους όρους του κειμένου και έχει μικρό μέγεθος, γιατί αναμένουμε ότι η σχέση δύο όρων φθίνει με την απόσταση. Τυπικά, δεν υπάρχει κάποιος λόγος για μεταβλητό μέγεθος παραθύρου, αν θέλουμε να υπάρχει ομοιομορφία στο πως αντιμετωπίζονται οι διάφοροι όροι μέσα στο κείμενο. Από την άλλη, είναι λογικό ανάλογα με το είδος του κειμένου, κάθε περιοχή να έχει διαφορετική σημασία για το πρόβλημα που αντιμετωπίζουμε. Ενδέχεται για παράδειγμα στα ειδησεογραφικά κείμενα η αρχή των κειμένων να είναι πολύ σημαντική για την ταξινόμηση ή για την σύνοψη, ενώ οι λέξεις κλειδιά να είναι πιο συχνές στη μέση ενός κειμένου, οπότε και πρέπει να δώσουμε βάρος στις αντίστοιχες περιοχές. Αν επομένως αυξηθεί το μέγεθος παραθύρου σε κάποιες περιοχές, τότε οι αντίστοιχοι κόμβοι/όροι θα έχουν περισσότερες ακμές και θα κυριαρχούν των υπολοίπων σε κριτήρια κεντρικότητας, βαθμού και σημασίας μέσα στο γράφο. Το που θα πρέπει να δωθεί έμφαση εξαρτάται από το είδος των κειμένων και από το πρόβλημα.

Κάθε όρος, και πιο συγκεκριμένα κάθε εμφάνιση του όρου, θα πρέπει να έχει το δικό του μέγεθος παραθύρου, το οποίο μπορεί να εξαρτάται από το ίδιο τον όρο, από τη συχνότητα εμφάνισης του ή και από τη θέση που βρίσκεται μέσα στο κείμενο. Το επεκτεταμένο αυτό μοντέλο γράφων λέξεων μας προσφέρει τη δυνατότητα να επικεντρωνόμαστε σε σημεία του κειμένου και σε όρους που πιθανώς να έχουν μεγαλύτερη σημασία, βασιζόμενοι σε ευριστικές ή αυτοματοποιημένες μεθόδους.

Τέλος, δεν πρέπει να ξεχνάμε ότι οι αλλαγές που γίνονται σε ένα σημείο του κειμένου δεν είναι τοπικές, αλλά επηρεάζουν και τα επόμενα κομμάτια. Για να γίνει πιο κατανοητό πως το διαφορετικό μήκος παραθύρου σε κάποια περιοχή επηρεάζει και τις υπόλοιπες, στο παρακάτω διάγραμμα φαίνεται και πάλι ο βαθμός του κάθε όρου σε συνάρτηση με τη θέση του στο κείμενο για το απλό μοντέλο, για το μοντέλο με Rebase και για το μοντέλο με μεταβλητό μέγεθος παραθύρου. Το κείμενο πρόκειται για το ίδιο που χρησιμοποιήθηκε στην προηγούμενη ενότητα και ο γράφος είναι κατευθυνόμενος, χωρίς βάρος, με μέγεθος παραθύρου 4 και οι πρώτοι 3 όροι έχουν τριπλάσιο μέγεθος παραθύρου:



Εικόνα 3.3: Επιρροή του μεταβλητού μεγέθους παραθύρου στο βάρος των όρων

Η κατάσταση είναι πολύ διαφορετική στα τρία μοντέλα. Το μοντέλο με Rebase δίνει το ίδιο βάρος σε όλους τους όρους. Στο μοντέλο μεταβλητού παραθύρου το διπλάσιο μήκος παραθύρου αύξησε τη σημασία των πρώτων τριών όρων, αλλά το κλιμακωτό φαινόμενο που παρατηρείται στο απλό μοντέλο υπάρχει και στο μοντέλο μεταβλητού παραθύρου και μάλιστα τώρα εμφανίζεται σε δύο σημεία. Στους τρεις τελευταίους όρους το βάρος είναι το ίδιο, αλλά στη μέση του κειμένου το βάρος είναι αυξημένο στο μοντέλο μεταβλητού παραθύρου σε σχέση με τα άλλα δύο. Αυτό συμβαίνει, γιατί αν και η αύξηση του μεγέθους περιορίζεται στην αρχή, επηρεάζονται και οι επόμενοι όροι/κόμβοι, αφού αυξήθηκαν σε αυτούς οι εισερχόμενες ακμές. Για να αυξηθούν οι εξερχόμενες ακμές σε κάποιο σημείο πρέπει αναγκαστικά να αυξηθούν και οι εισερχόμενες σε κάποιο άλλο.

3.6 Ensembles Γράφων Λέξεων

Η μέθοδος ensemble στο χώρο της μηχανικής μάθησης είναι ο συνδυασμός πολλών και διαφορετικών μοντέλων με τέτοιο τρόπο ώστε να παραχθούν συνολικά καλύτερες προβλέψεις σε σχέση με τα επιμέρους μοντέλα και είναι μια δημοφιλής μέθοδος στον τομέα για την βελτίωση της απόδοσης.

Για να γίνει πιο κατανοητή η λειτουργία των ensembles ας θεωρήσουμε μια ομάδα ανθρώπων η οποία καλείται να λάβει μια απόφαση. Τα μέλη της ομάδας μπορεί να έχουν όλα παρόμοιες γνώσεις ή μπορεί να υπάρχουν και μερικοί οι οποίοι είναι πιο ειδικοί για το πρόβλημα που καλούνται να λάβουν απόφαση. Οι τρόποι με τους οποίους τα μέλη συνεργάζονται για να λάβουν την τελική απόφαση μπορεί να είναι πολλοί: μπορεί να γίνει μια ψηφοφορία όπου κάθε μέλος έχει μια ψήφο και η πλειοψηφία κερδίζει ή ενδέχεται οι

ειδικοί να έχουν περισσότερες ψήφους και άρα βαρύτητα ή μπορεί τα μέλη να έχουν το δικαίωμα βέτο ή το δικαίωμα να αποφασίσει μόνο του ένα άτομο, εφόσον είναι σίγουρο για αυτήν την απόφαση και τον εμπιστεύεται όλη η ομάδα. Στη γενική μορφή, η απόφαση μπορεί να παρθεί από έναν εξωτερικό κριτή, ο οποίος συγκεντρώνει τις απόψεις όλων των μελών της ομάδας και με βάση την αξιοπιστία των μελών, το είδος του προβλήματος, τις παλαιότερες αποφάσεις και άλλα κριτήρια, παίρνει μια τελική απόφαση. Σε αυτή την περίπτωση, φαίνεται φυσιολογικό η συλλογική απόφαση να είναι καλύτερη από τις ατομικές, με προϋπόθεση βέβαια ότι οι γνώσεις όλων των μελών συνδυάζονται με το σωστό τρόπο.

Αν τώρα αντικαταστήσουμε τους ανθρώπους με μοντέλα μηχανικής μάθησης και τις αποφάσεις με τις προβλέψεις των μοντέλων, καταλήγουμε στον ορισμό των ensembles. Επιπλέον, το κριτήριο της απόφασης μπορεί να βασίζεται στην έξοδο των μοντέλων ή στις πιθανότητες που δίνουν τα μοντέλα σε κάθε κλάση, ενώ γενικά ο εξωτερικός κριτής είναι ένα διαφορετικό μοντέλο μηχανικής μάθησης το οποίο μαθαίνει από τις προβλέψεις των υπολοίπων. Η μέθοδος ensemble είναι στην πραγματικότητα μια οικογένεια μεθόδων, αφού ο κάθε διαφορετικός συνδυασμός των προβλέψεων των μοντέλων, μας δίνει και μια νέα ensemble, μια νέα ομάδα. Η χρησιμότητα της μεθόδου προκύπτει από το γεγονός ότι το νέο μοντέλο δεν περιορίζεται από τις ιδιοσυγκρασίες των επιμέρους μοντέλων και από τα λάθη που μπορεί να κάνουν, οπότε μπορεί να πετύχει μεγαλύτερη γενίκευση.

Βέβαια, ένα σημαντικό κριτήριο για να μπορεί η μέθοδος αυτή να είναι χρήσιμη, είναι να μην έχουν συσχέτιση μεταξύ τους τα λάθη των επιμέρους μοντέλων. Αν όλοι οι άνθρωποι κάνουν τα ίδια λάθη, τότε η ομάδα δεν μπορεί να λάβει καλύτερες αποφάσεις, αλλά θα κάνει και αυτή τα ίδια λάθη. Είναι χρήσιμο λοιπόν κάποιες φορές τα άτομα να έχουν διαφορετικές απόψεις, ώστε να αναδεικνύονται διαφορετικές επιλογές και να υπάρχει ποικιλία που θα οδηγήσει με τη σειρά της στη καλύτερη γενίκευση. Όμως από την άλλη, πρέπει να προσέξουμε να μην εισάγουμε πολύ μεγάλη ποικιλία στις αποφάσεις, καθώς αυτό μπορεί να βλάψει τη συνολική απόδοση. Δεν αρκεί να έχουμε διαφορετικές επιλογές, αλλά θα πρέπει αυτές να προέρχονται και από ένα αξιόπιστο μέλος το οποίο δεν κάνει σχεδόν τυχαίες επιλογές. Έτσι και στα ensembles, αν όλα τα μοντέλα κάνουν λάθη σε παρόμοιες εισόδους, δεν θα υπάρχει κάποια βελτίωση στην απόδοση και ταυτόχρονα η ποικιλία δεν θα πρέπει να προκύψει εις βάρος της ποιότητας των μοντέλων.

Η χρήση ensembles έχει μεγαλύτερο πρακτικό ενδιαφέρον από ότι ακαδημαϊκό, αφού οποιαδήποτε καινοτομία αφορά απλά τον τρόπο που θα συνδυαστούν τα επιμέρους μοντέλα και τα χαρακτηριστικά που πρέπει να έχουν αυτά. Ωστόσο, στη περίπτωση των γράφων λέξεων, η μέθοδος προσφέρεται για άμεση βελτίωση της απόδοσης, λόγω της ποικιλίας των γράφων και των διαφόρων επιλογών που έχουμε για την κατασκευή τους. Οι γράφοι μπορεί να είναι κατευθυνόμενοι ή μη, με βάρος ή χωρίς και το μέγεθος παραθύρου μπορεί να λάβει διάφορες τιμές. Επιπλέον, υπάρχει η δυνατότητα να εφαρμοστούν και οι μέθοδοι ενίσχυσης κόμβων και μεταβλητού παραθύρου, ώστε να λάβουμε διάφορα μοντέλα που να δίνουν έμφαση σε διαφορετικό μέρος του κειμένου.

Συνολικά λοιπόν, αν και το βασικό μοντέλο είναι μόνο ένα και τα κείμενα είναι τα ίδια, οι γράφοι που μπορούν να κατασκευαστούν είναι πολλοί και με αρκετή ποικιλία, ώστε η μέθοδος ensemble γράφων λέξεων να είναι αρκετά χρήσιμη.

Κεφάλαιο 4

Σχεδιασμός και Υλοποίηση

Στο κεφάλαιο αυτό αναπτύσσονται τα τεχνικά θέματα υλοποίησης των μεθόδων που παρουσιάστηκαν στο Κεφάλαιο 3 και τα σύνολα δεδομένων που χρησιμοποιήθηκαν. Αρχικά, περιγράφονται τα εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη του μοντέλου γράφων λέξεων και οι τεχνικές λεπτομέρειες για την υλοποίηση των προτεινόμενων τροποποιήσεων για το μοντέλο. Τέλος, γίνεται αναφορά στα σύνολα δεδομένων που χρησιμοποιήθηκαν για την αξιολόγηση.

4.1 Εργαλεία

Η ανάπτυξη και υλοποίηση του μοντέλου γράφων λέξεων έγινε στη γλώσσα προγραμματισμού Python, σε αντιστοιχία με το μοντέλο TF-IDF της βιβλιοθήκης scikit-learn [18]. Τα δύο βασικά κομμάτια της υλοποίησης είναι η προεπεξεργασία των κειμένων και η κατασκευή του γράφου, οπότε ήταν σημαντική η επιλογή μιας γλώσσας που να επιταχύνει αυτές τις διαδικασίες και να περιέχει βιβλιοθήκες με πολλές επιλογές και δυνατότητες για τις λειτουργίες που χρειάζονται. Για την προεπεξεργασία των κειμένων χρησιμοποιήθηκε η βιβλιοθήκη spacy [19] καθώς έχει και δυνατότητα για coreference resolution μέσω της επέκτασης της NeuralCoref[1], ενώ για το collocation detection και για την παραγωγή word embeddings χρησιμοποιήθηκε η βιβλιοθήκη gensim [20]. Το κυριότερο bottleneck από άποψη ταχύτητας για την χρήση των γράφων λέξεων είναι η κατασκευή των γράφων. Η βιβλιοθήκη NetworkX [21] προσφέρει εύκολη κατασκευή και διαχείριση των γράφων και είναι πολύ χρήσιμη για εξειδικευμένες λειτουργίες πάνω στο γράφο. Από την άλλη πλευρά, υστερεί σε θέματα ταχύτητας και είναι πολύ απαιτητική στη μνήμη που χρειάζεται για τη κατασκευή και αποθήκευση των γράφων, οπότε τελικά προτιμήθηκε η βιβλιοθήκη python-igraph[2]. Τέλος, για την ταξινόμηση των κειμένων και την αξιολόγηση των γράφων λέξεων χρησιμοποιήθηκε η δημοφιλής βιβλιοθήκη μηχανικής μάθησης scikit-learn.

4.2 Υλοποίηση

Coreference Resolution

Η βιβλιοθήκη NeuralCoref χρησιμοποιεί τεχνητά νευρωνικά δίκτυα και word embeddings τα οποία έχουν προπονηθεί στη συλλογή κειμένων OntoNotes 5.0 [22] και βασίζεται στις εργασίες [23], [24]. Το γεγονός αυτό επιτρέπει την προπονήση του μοντέλου σε οποιοδήποτε σύνολο κειμένων, όμως για να γίνει αυτό πρέπει να ξέρουμε ήδη τα coreference που

[1] https://github.com/huggingface/neuralcoref
[2] https://igraph.org/python/

υπάρχουν στα κείμενα, κάτι που καταρρίπτει το λόγο χρήσης του μοντέλου. Αν ήταν γνωστά τα coreferences των κειμένων δεν θα υπήρχε ανάγκη του μοντέλου για την εύρεση τους. Η γενίκευση που μπορεί να παρέχει ένα μοντέλο της μηχανικής μάθησης εξαρτάται από τη συνέπεια και την ομοιότητα που έχουν μεταξύ τους τα δεδομένα στα οποία έχει προπονηθεί και τα δεδομένα στα οποία το εφαρμόζουμε. Με άλλα λόγια, οι κανόνες για coreference resolution που έχει μάθει το μοντέλο μπορεί να μην είναι κατάλληλοι για όλα τα είδη των κειμένων και αυτό είναι κάτι που πρέπει να θυμόμαστε πριν εφαρμόσουμε το μοντέλο σε διαφορετική συλλογή κειμένων.

Στο πλαίσιο αυτό, επιλέχθηκε να κρατήσουμε μόνο τις απλές αναφορές όπως Noun to Noun, Pronoun to Noun και γενικά coreferences από spans μεγέθους μέχρι N λέξεων σε spans μέχρι M λέξεων, όπου N,M μικροί φυσικοί αριθμοί και N<M. Το span είναι ένα συνεχές διάστημα λέξεων του κειμένου, απ' όπου μπορεί να ξεκινά ή να καταλήγει ένα coreference. Η λογική πίσω από αυτό, είναι ότι οι αναφορές που περιέχουν πολλούς όρους δεν είναι συνήθως σωστές ή ιδιαίτερα χρήσιμες, ενώ αν περιοριστούμε μόνο στις πολύ απλές αναφορές δεν θα κάνουμε αρκετές αλλαγές στο κείμενο, ώστε να υπάρχει διαφορά στην απόδοση. Τέλος, το επεξεργασμένο κείμενο με τα coreferences προκύπτει απλά από την αντικατάσταση των αναφορικών όρων από τους όρους στους οποίους αναφέρονται.

Collocation detection

Για το collocation detection χρησιμοποιήθηκε η μέθοδος που προτείνεται στο [25]. Πρόκειται για ένα απλό στατιστικό μοντέλο το οποίο βασίζεται στη συχνότητα των unigram (μονο-λέξεων) και bigram (δι-λέξεων) ως εξής:

$$score(w_i, w_j) = \frac{count(w_i w_j) - \min\_count}{count(w_i) * count(w_j)}$$

Η μέθοδος έχει δύο παραμέτρους που πρέπει να οριστούν, το min_count και το $threshold$. Το min_count είναι ένας παράγοντας που αποτρέπει τον σχηματισμό πολλών εκφράσεων που δεν εμφανίζονται πολύ συχνά. Αν το score για δύο όρους w_i, w_j υπερβαίνει το $threshold$ τότε οι δύο όροι αυτοί αποτελούν ένα collocation. Το πρόβλημα με αυτήν τη μέθοδο είναι ότι το score δεν έχει ανώτατο όριο στη τιμή που μπορεί να πάρει και το εύρος εξαρτάται από το μέγεθος και το είδος της συλλογής κειμένων που έχουμε. Για να αντιμετωπιστεί αυτό το πρόβλημα στο [26] προτείνεται μια εναλλακτική μέθοδος για την ανάθεση τιμής σε ένα bigram, ώστε αυτή να είναι κανονικοποιημένη στο εύρος [-1,1] και βασίζεται στη μετρική PMI (Pointiwise Mutual Information) από τη θεωρία πληροφοριών:

$$score(w_i, w_j) = \frac{\ln\left(\frac{prob(w_i, w_j)}{prob(w_i) * prob(w_j)}\right)}{-\ln\left(prob(w_i, w_j)\right)}, \qquad \text{όπου } prob(w) = \frac{count(w)}{corpus\_word\_count}$$

Για να πάρουμε το επεξεργασμένο κείμενο, κάθε bigram collocation που εντοπίζουμε το αντικαθιστούμε με ένα όρο της μορφή όρος1_όρος2 και η διαδικασία αυτή μπορεί να επαναληφθεί 2-4 φορές για να πάρουμε όλο και μεγαλύτερες εκφράσεις. Για παράδειγμα, στο πρώτο πέρασμα μπορεί να σχηματιστεί η φράση *new_york* από τους όρους *new* και *york* και στο δεύτερο η φράση *new_york_city* από τον καινούργιο όρο *new_york* και από τον όρο *city*.

Word Embeddings

Η αναπαράσταση των όρων ως διανύσματα μας επιτρέπει να συγκρίνουμε τους όρους αυτούς και να δούμε πόσο όμοιοι ή διαφορετικοί είναι μέσω της απόστασης ομοιότητας των διανυσμάτων. Η απόσταση υπολογίζεται ως το cosine distance των word embeddings των όρων και παίρνει τιμές στο εύρος [-1,1], αφού μας ενδιαφέρει το πόσο όμοια είναι τα δύο διανύσματα και όχι το πόσο απέχουν μεταξύ τους στο διανυσματικό χώρο. Ο γενικός τύπος για την ανάθεση βάρους στις ακμές με χρήση των word embeddings είναι:

$$w_{i,j} = a + b * f\left(sim\left(emb_i, emb_j\right)\right)$$

όπου $a, b \in [0,1]$, $sim\left(emb_i, emb_j\right) = \frac{\boldsymbol{emb_i} \cdot \boldsymbol{emb_j}}{\|\boldsymbol{emb_i}\| * \|\boldsymbol{emb_j}\|}$ και f μια συνάρτηση εκ των $\max(x, 0)$, $abs(x)$, $identity(x)$.

Αν θέλουμε να λάβουμε υπόψη την πολλαπλότητα μιας ακμής μπορούμε, σε αναλογία με τους απλούς γράφους με βάρος, να προσθέσουμε ξανά το βάρος σε κάθε επανάληψη της ακμής. Οι μεταβλητές a και b ρυθμίζουν την επιρροή της συνύπαρξης και της ομοιότητας αντίστοιχα. Αν $b = 0$ τότε δεν λαμβάνουμε υπόψη καθόλου τη απόσταση των όρων και βασιζόμαστε μόνο στη συνύπαρξη, όπως στους κλασικούς γράφους λέξεων. Ομοίως, αν $a = 0$ βασιζόμαστε μόνο στην ομοιότητα των όρων για την ανάθεση βάρους. Βέβαια, πρέπει να θυμόμαστε ότι οι ακμές προκύπτουν πάντα με βάση τη συνύπαρξη των όρων σε ένα παράθυρο, οπότε ακόμα και τότε η συνύπαρξη παίζει κάποιο ρόλο, αφού καθορίζει ποιες ακμές θα δημιουργηθούν στο γράφο.

Η συνάρτηση f χρησιμοποιείται, γιατί η απόσταση δύο όρων μπορεί να είναι και αρνητική σε περίπτωση που οι δύο όροι είναι ανόμοιοι, αλλά όχι ασυσχέτιστοι. Ειδικότερα, αν δύο διανύσματα έχουν cosine distance ίση με 0, τότε τα διανύσματα είναι κάθετα μεταξύ τους και οι αντίστοιχοι όροι δεν έχουν κάποια σχέση μεταξύ τους. Αν τώρα δύο διανύσματα έχουν cosine distance 1 ή -1 τότε τα διανύσματα δείχνουν προς την ίδια ή αντίθετη κατεύθυνση και σχετίζονται θετικά ή αρνητικά αντίστοιχα. Ενδέχεται λοιπόν η πληροφορία ότι δύο όροι σχετίζονται αρνητικά να είναι περιττή ή αντίθετα να είναι χρήσιμη μόνο αν υπάρχει κάποια σχέση και όχι αν αυτή είναι αρνητική ή θετική. Στην πρώτη περίπτωση η συνάρτηση $\max(x, 0)$ δεν λαμβάνει υπόψη αρνητικές συσχετίσεις, ενώ στη δεύτερη περίπτωση η $abs(x)$ δίνει το ίδιο βάρος στις αρνητικές και θετικές συσχετίσεις.

Στο πλαίσιο αυτό, μελετήθηκαν διάφοροι συνδυασμοί των μεταβλητών a, b και της συνάρτησης f ώστε να εξεταστεί αν η απόσταση ομοιότητας των όρων μπορεί να βελτιώσει την ανάθεση βάρους στις ακμές. Τα word embeddings μπορεί είτε να έχουν ήδη προπονηθεί σε κάποια άλλη συλλογή κειμένων είτε να προπονηθούν από την αρχή στη συλλογή κειμένων που θα χρησιμοποιηθεί. Προτιμήθηκε η δεύτερη επιλογή, γιατί συνήθως είναι πιο χρήσιμο η εκπαίδευση να έχει γίνει πάνω σε σχετικά κείμενα και το μέγεθος των συνόλων δεδομένων που χρησιμοποιήθηκαν δεν είναι τόσο μεγάλο, ώστε η διαδικασία αυτή να είναι πολύ χρονοβόρα.

Ενίσχυση κόμβων

Η άνιση αντιπροσώπευση των πρώτων και τελευταίων όρων των κειμένων από τους γράφους λέξεων είναι ένα πρόβλημα που προκύπτει από τον ορισμό του μοντέλου και δεν υπάρχει προφανής τρόπος για να ξεπεραστεί. Όπως φαίνεται στο διάγραμμα 3.3, οι ανεπηρέαστοι όροι έχουν $ws - 1$ εισερχόμενες ακμές και $ws - 1$ εξερχόμενες ακμές, έτσι ώστε συνολικά να έχουν βάρος $2 * (ws - 1)$. Το ίδιο ισχύει και στους μη-κατευθυνόμενους

γράφους χωρίς βάρος, καθώς ένας ανεπηρέαστος όρος θα έχει $2 * (ws - 1)$ γείτονες. Για το λόγο αυτό, στην μέθοδο Rebase ορίζεται ένα όριο $threshold$, κάτω από τα οποίο τα βάρη δεν επιτρέπεται να είναι:

$$weight(v) = \max(degree(v), threshold)$$
$$threshold = min\ (2 * (ws - 1), |d| - 1)$$

Με ws συμβολίζεται το μέγεθος του παραθύρου και $|d|$ το μήκος του κειμένου. Η μέγιστη τιμή που μπορεί να πάρει το $threshold$ είναι $|d| - 1$, αφού τόσες είναι οι μέγιστες συνδέσεις που μπορεί να σχηματίσει θεωρητικά ένας όρος στο κείμενο d.

Στη μέθοδο Boost ενδιαφερόμαστε μόνο για τους ακριανούς όρους. Για να αυξηθεί το βάρος στους πρώτους $ws - 1$ όρους έχουμε:

$$\text{weight}(v_1) = \deg(v_1) + ws - 1$$
$$\text{weight}(v_2) = \deg(v_2) + ws - 2$$
$$\dots$$
$$\text{weight}(v_{ws-1}) = \deg(v_{ws-1}) + 1$$

ενώ αντίστοιχα για τους τελευταίους $ws - 1$ όρους:

$$\text{weight}\left(v_{|d|}\right) = \deg\left(v_{|d|}\right) + ws - 1$$
$$\text{weight}\left(v_{|d|-1}\right) = \deg\left(v_{|d|-1}\right) + ws - 2$$
$$\dots$$
$$\text{weight}\left(v_{|d|-(ws-2)}\right) = \deg\left(v_{|d|-(ws-2)}\right) + 1$$

όπου ο κόμβος v_i αντιστοιχεί στον i-οστό όρο του κειμένου.

Το βάρος του πρώτου και του τελευταίου όρου του κειμένου αυξάνεται κατά $ws - 1$, ώστε να προσομοιωθεί η υπάρξη $ws - 1$ εισερχόμενων και εξερχόμενων ακμών αντίστοιχα. Ομοίως αντίστοιχες αλλαγές πραγματοποιούνται και για τους υπόλοιπους όρους που έχουν αδικηθεί.

Για να προκύψουν οι παραπάνω προσεγγίσεις έχουμε υποθέσει ένα ιδεατό κείμενο που αποτελείται από ξεχωριστούς όρους. Ωστόσο, στα πραγματικά κείμενα τα βάρη των ανεπηρέαστων όρων μπορεί να είναι μικρότερα, αφού στους γράφους χωρίς βάρη μερικές ακμές επαναλαμβάνονται και δεν συνεισφέρουν στο βαθμό. Η συχνότητα της επανάληψης ακμών εξαρτάται από το ίδιο το κείμενο και από τη χρήση κατευθυνόμενων ή μη ακμών. Στους μη κατευθυνόμενους γράφους η επανάληψη μιας ακμής είναι πιο εύκολη, γιατί δεν υπάρχει ο διαχωρισμός σε εξερχόμενες και εισερχόμενες ακμές, όπως στους κατευθυνόμενους γράφους. Σε αυτή την περίπτωση, εάν ένας όρος επαναλαμβάνεται πριν και μετά από ένα άλλο, θα σχηματιστούν δύο ακμές, μια εξερχόμενη και μια εισερχόμενη, ενώ στη μη κατευθυνόμενη περίπτωση θα σχηματιστεί μόνο μια ακμή που θα συνδέει αυτούς τους όρους. Αν θέλουμε να λάβουμε υπόψη τις επαναλήψεις μπορούμε είτε να διαλέξουμε ένα ελαφρώς πιο μικρό όριο στην πρώτη προσέγγιση είτε να προσθέσουμε μικρότερο βάρος στη δεύτερη προσέγγιση. Στη εργασία αυτή επιλέχθηκε να μην ληφθούν υπόψη οι επαναλήψεις και να αυξήσουμε το βάρος των όρων θεωρώντας το μέγιστο αριθμό νέων ακμών που τους αναλογεί, γιατί δεν προέκυψαν σημαντικές διαφορές στα αποτελέσματα που να δικαιολογούν τη μια ή την άλλη επιλογή.

Μεταβλητό μέγεθος παραθύρου

Στη μέθοδο μεταβλητού παραθύρου, σκοπός είναι να δοθεί μεγαλύτερη έμφαση και προσοχή σε διαφορετικούς όρους του κειμένου, όπου κάθε όρος θα έχει το δικό του μέγεθος παραθύρου ανάλογα με τη θέση που έχει στο κείμενο αλλά και τον ίδιο τον όρο. Είναι φυσιολογικό λοιπόν να υπάρχουν πολλές επιλογές για το πως θα γίνει αυτή η ανάθεση, σε ποιους όρους θα δοθεί έμφαση και με ποια κριτήρια. Για παράδειγμα, μπορεί να είναι χρήσιμο να δοθεί μεγαλύτερο βάρος στους όρους που εμφανίζονται σπάνια σε άλλα κείμενα αλλά συχνά στο παρόν (δηλαδή με υψηλό idf) ή στους όρους που περιέχουν αριθμούς ή στους όρους που εμφανίζονται στην αρχή του κειμένου. Η πολυπλοκότητα γίνεται ακόμα μεγαλύτερη, αν θεωρήσουμε ότι η ανάθεση των μεγεθών των παραθύρων γίνεται με διαφορετικό τρόπο σε κάθε κείμενο της συλλογής, όπου κάθε εμφάνιση ενός όρου θα πρέπει να συνοδεύεται και από το δικό της μέγεθος παραθύρου.

Στη εργασία αυτή βασιστήκαμε σε πρώτη φάση σε ευριστικές μεθόδους που λαμβάνουν υπόψη μόνο τη σειρά των όρων μέσα στο κείμενο και δίνουν έμφαση κυρίως στα άκρα του κειμένου, τα οποία αδικούνται από το παραδοσιακό μοντέλο γράφων λέξεων. Η ανάγκη για την ύπαρξη μιας πιο αυτοματοποιημένης μεθόδου είναι εμφανής και θα συζητηθεί στο Κεφάλαιο 6.

Ensemble

Ο στόχος στη παρούσα εργασία δεν είναι να μελετηθούν οι διάφοροι μέθοδοι ensemble που υπάρχουν ή να προταθεί μια καινούργια, αλλά ένα proof of concept για τη χρησιμότητα και τη καταλληλότητα των μεθόδων αυτών στους γράφους λέξεων. Όπως αναφέρθηκε, για να είναι χρήσιμα τα ensembles πρέπει να υπάρχει ποικιλία στις προβλέψεις και η οποία μπορεί να προέρχεται είτε από διαφορετικούς αλγόριθμους μάθησης και διαφορετικούς ταξινομητές (classifiers) είτε από διαφορετική προεπεξεργασία των δεδομένων. Στην περίπτωση των γράφων λέξεων, η ποικιλία προκύπτει από την διαφορετική διανυσματική αναπαράσταση των κειμένων μέσω των διαφόρων επιλογών που υπάρχουν για την κατασκευή των γράφων, όπως το μέγεθος παραθύρου, η κατεύθυνση και το βάρος. Η ensemble θα αποτελείται μόνο από γράφους λέξεων και θα χρησιμοποιείται πάντα ο ίδιος classifier, αλλά η ποικιλία που εισάγουμε μέσω ουσιαστικά της διαφορετικής επεξεργασίας των κειμένων είναι ικανή να μας δώσει καλύτερα αποτελέσματα.

Οι μέθοδοι ensemble που θα εξεταστούν είναι η ψηφοφορία με πλειοψηφία (hard vote), η ψηφοφορία με χρήση των πιθανοτήτων που δίνονται σε κάθε κλάση (soft vote) και το stacking. Στην ψηφοφορία με πλειοψηφία, κάθε μοντέλο λαμβάνει μια ή περισσότερες ψήφους για τη κλάση στην οποία ανήκει το κείμενο και η κλάση με τις περισσότερες ψήφους επιλέγεται ως η απόφαση του ensemble. Στη δεύτερη περίπτωση, ο classifier δίνει ως έξοδο τις πιθανότητες που αναθέτει για κάθε κλάση και η τελική απόφαση λαμβάνεται με βάση τον μέσο όρο αυτών των πιθανοτήτων. Το stacking ensemble χρησιμοποιεί ένα μετα-μοντέλο το οποίο προβλέπει βάση των προβλέψεων των απλών μοντέλων. Με αυτόν τον τρόπο, το μετα-μοντέλο μπορεί να μάθει ποια μοντέλα είναι χρήσιμα και σε ποιες περιπτώσεις και έτσι μπορεί να παράγει καλύτερες προβλέψεις. Ποιο θα είναι το μετα-μοντέλο, πως ακριβώς θα προπονηθεί ή ποια και πόσα θα είναι τα απλά μοντέλα, είναι ζητήματα που πρέπει να προσδιοριστούν ανά περίπτωση και δεν υπάρχει γενική λύση. Στα πειράματα του 5^{ου} Κεφαλαίου, το μετα-μοντέλο θα είναι το Logistic Regression το οποίο θα προπονείται ως εξής:

- Χωρίζουμε τα κείμενα σε K-folds. Για κάθε ένα fold προπονούμε N βασικά μοντέλα στα υπόλοιπα folds και παράγουμε προβλέψεις για το ισχύον. Στο τέλος της

διαδικασίας, για κάθε ένα κείμενο θα έχουμε παράγει τις προβλέψεις Ν διαφορετικών μοντέλων και οι οποίες όλες μαζί θα αποτελούν το train set για το μετα-μοντέλο, δηλαδή το meta train set.

- Προπονούμε τα απλά μοντέλα σε όλο το training set και παράγουμε προβλέψεις για το test set. Το σύνολο των προβλέψεων θα αποτελέσει το meta test set για το μετα-μοντέλο.
- Προπονούμε το μετα-μοντέλο στο meta train set του πρώτου βήματος και παράγουμε νέες προβλέψεις βάση του meta test set του δεύτερου βήματος.

Οι προβλέψεις που παράγουν τα απλά μοντέλα είναι οι πιθανότητες που αποδίδουν σε κάθε κλάση για κάθε κείμενο. Επιπλέον, αντί να εκπαιδεύουμε ξανά τα απλά μοντέλα στο δεύτερο βήμα, το meta test set μπορεί να προκύψει ως ο μέσος όρος των προβλέψεων των απλών μοντέλων που έχουν προπονηθεί στα folds.

4.3 Σύνολα δεδομένων

Τα σύνολα δεδομένων που χρησιμοποιήθηκαν για την αξιολόγηση των προτεινόμενων τεχνικών για τους γράφους λέξεων είναι το 20 Newsgroups[3] (20NG) και το Reuters8 (R8) [27]. Για λόγους που θα εξηγηθούν στην επόμενη ενότητα, στο 20NG έχουν αφαιρεθεί όλα τα κείμενα που περιέχουν λιγότερους από 50 χαρακτήρες, εκ των οποίων 533 ανήκουν στο σύνολο προπόνησης και 386 στο σύνολο αξιολόγησης. Τα δύο αυτά σύνολα δεδομένων είναι κλασικά στο χώρο της ταξινόμησης κειμένων και είναι ήδη χωρισμένα σε σύνολα εκπαίδευσης (train set) και σύνολα αξιολόγησης (test set). Το R8 είναι ένα εύκολο σύνολο για κατηγοριοποίηση και η απόδοση που πετυχαίνουν ακόμα και απλά μοντέλα είναι αρκετά μεγάλη, οπότε οποιαδήποτε βελτίωση είναι σχετικά δύσκολή. Από την άλλη, το 20NG είναι πιο δύσκολο συγκριτικά και υπάρχει μεγαλύτερος χώρος για βελτίωση. Τα κείμενα του R8 είναι αρκετά σύντομα, έχουν γραφτεί από αρθρογράφους, αλλά οι κλάσεις δεν είναι ισοκατανεμημένες, ενώ στο 20NG τα κείμενα έχουν γραφτεί από χρήστες του διαδικτύου, το μέγεθος των κειμένων ποικίλει αρκετά και η πληθικότητα των κλάσεων είναι περίπου η ίδια. Στη συνέχεια περιγράφονται πιο αναλυτικά τα σύνολα δεδομένων που χρησιμοποιήθηκαν.

	# Docs	# Train Docs	# Test Docs	Classes	# unique terms	Avg # Terms	Avg # Vertices	Avg # Edges
20NG	17,927	10,781	7,146	20	128,845	179	88	452
R8	7,674	5,485	2,189	8	19,820	98	54	252

Πίνακας 4.1: Στατιστικά των συλλογών κειμένου 20 Newsgroups και Reuters8

4.3.1 20 Newsgroups

Το 20NG προέρχεται από δημοσιεύσεις σε 20 διαφορετικές ομάδες συζητήσεων (online newsgroups) που καθεμιά αφορά κάποιο συγκεκριμένο θέμα. Οπως για τα περισσότερα σύνολα δεδομένων που είναι διαθέσιμα στο διαδίκτυο, έτσι και για το 20NG, υπάρχουν

[3] https://scikit-learn.org/0.19/datasets/twenty_newsgroups.html

διαθέσιμες πολλές διαφορετικές εκδόσεις. Οι διαφορές μεταξύ των εκδόσεων δεν είναι μεγάλες, αλλά για τυπικούς λόγους θα αναφερθούμε σύντομα στα διαθέσιμα σύνολα δεδομένων σχετικά με το 20NG.

Η πρώτη έκδοση περιέχει συνολικά 19997 κείμενα, εκ των οποίων μερικά επαναλαμβάνονται. Επιπλέον, τα κείμενα αυτά περιέχουν πολλές μετα-πληροφορίες για τα ίδια τα κείμενα, όπως οι ηλεκτρονικές διευθύνσεις των χρηστών, σε ποιο newsgroup ανήκει, η ημερομηνία, το NNTP-Posting-Host, πόσες γραμμές είναι το κείμενο και άλλα. Επειδή λοιπόν οι πληροφορίες αυτές μπορεί είναι πολύ χαρακτηριστικές για το είδος των κείμενων, έχουν προκύψει επιμέρους εκδόσεις του 20NG οι οποίες αφαιρούν κάποιες ή όλες τις επικεφαλίδες που περιέχουν αυτές τις πληροφορίες. Για παράδειγμα, η έκδοση bydate σπάει το 20NG σε train και test υποσύνολα με βάση τη μέρα έκδοσης των δημοσιεύσεων και αφαιρεί από την επικεφαλίδα τα πεδία Xref, Newsgroups, Path, Followup-To και Date. Ωστόσο, ακόμα και τότε, η επικεφαλίδα περιέχει πολλές πληροφορίες τις οποίες κάποιος ταξινομητής μπορεί να εκμεταλλευτεί, όπως το όνομα οργανισμών ή πανεπιστημίων ή ακόμα και οι ηλεκτρονικές διευθύνσεις κάποιων μελών που συμμετείχαν πολύ ενεργά στις συζητήσεις. Επομένως, για να έχουμε μια πιο ρεαλιστική κατηγοριοποίηση που να βασίζεται στο ίδιο το κείμενο, θα πρέπει να αφαιρεθούν όλες οι μετα-πληροφορίες από τις αρχικές δημοσιεύσεις.

Η έκδοση του 20NG που χρησιμοποιήθηκε στο Κεφάλαιο 5 προέρχεται από τη βιβλιοθήκη scikit-learn και περιέχει συνολικά 18,846 κείμενα, εκ των οποίων τα 11,314 ανήκουν στο σύνολο προπόνησης και τα υπόλοιπα 7,532 στο σύνολο αξιολόγησης. Από τις δημοσιεύσεις έχουν αφαιρεθεί οι επικεφαλίδες (headers), τα υποσέλιδα (footers) και οι απευθείας αναφορές άλλων δημοσιεύσεων (quotes). Εξαιτίας αυτής της αφαίρεσης, πολλά κείμενα έχουν μείνει κενά, περιέχουν μόνο κενές γραμμές ή χαρακτήρες και έχουν μικρύνει πολύ σε μέγεθος. Για το λόγο αυτό, επιλέχθηκε να αφαιρεθούν τα κείμενα τα οποία περιέχουν κάτω από 50 χαρακτήρες, τόσο από το σύνολο προπόνησης, όσο και από το σύνολο αξιολόγησης, οπότε και έμειναν 10,781 και 7,146 κείμενα αντίστοιχα και 17,927 συνολικά. Κάθε κείμενο ανήκει σε μια εκ των 20 κατηγοριών, οι οποίες φαίνονται στο πίνακα 4-1. Μερικές από αυτές είναι συγγενικές, όπως οι *comp.sys.ibm.pc.hardware* και η *comp.sys.ibm.pc.hardware*, ενώ κάποιες άλλες είναι σχετικά ξένες με τις υπόλοιπες, όπως η *misc.forsale*.

Label	# Train Docs	# Test Docs	Label	# Train Docs	# Test Docs
alt.atheism	458	303	**rec.sport.hockey**	565	385
comp.graphics	557	373	**sci.crypt**	576	368
comp.os.ms-windows.misc	558	375	**sci.electronics**	564	377
comp.sys.ibm.pc.hardware	574	378	**sci.med**	572	377
comp.sys.mac.hardware	547	364	**sci.space**	572	371
comp.windows.x	580	377	**soc.religion.christian**	586	384
misc.forsale	563	373	**talk.politics.guns**	529	343
rec.autos	547	365	**talk.politics.mideast**	526	363
rec.motorcycles	560	367	**talk.politics.misc**	445	297
rec.sport.baseball	550	371	**talk.religion.misc**	354	236

Πίνακας 4.2: Κατανομή των κλασεων στο 20 Newsgroups

4.3.2 Reuters8

Το σύνολο δεδομένων R8 είναι ένα υποσύνολο του Reuters21578. Το μεγαλύτερο αυτό σύνολο αποτελείται από 21,578 ειδησεογραφικά άρθρα τα οποία προέρχονται, όπως προδίδει το όνομα του, από το πρακτορείο Reuters το 1987 και αφορούν κυρίως οικονομικές και επιχειρησιακές ειδήσεις. Αν και αποτελείται συνολικά από 135 κατηγορίες κειμένων, όπως αποδόθηκαν από τον εκδότη, οι περισσότερες περιέχουν πολύ λίγα κείμενα, ενώ πολλά κείμενα ανήκουν σε παραπάνω από μια κατηγορία. Για το λόγο αυτό δημιουργήθηκαν τα υποσύνολα R8 και R52 κρατώντας μόνο τις 8 και 52 πιο συχνές κατηγορίες αντίστοιχα και τα οποία περιέχουν κείμενα τα οποία ανήκουν σε μια μόνο κατηγορία. Το R8 περιέχει συνολικά 7,674 κείμενα, εκ των οποίων τα 5,485 είναι κείμενα προπόνησης και τα 2,189 αξιολόγησης, ενώ αποτελείται φυσικά από 8 κατηγορίες. Όπως φαίνεται στον πίνακα 4.3, τα κείμενα δεν είναι σε καμιά περίπτωση ισοκατανεμημένα, αφού το 80% των κειμένων ανήκουν σε μία εκ των δύο κατηγοριών *acq* και *earn*, οπότε ακόμα και ένας απλός ταξινομητής μπορεί να πετύχει πολύ καλή απόδοση. Επιπλέον, αν και τα κείμενα είναι ειδησεογραφικά, είναι αρκετά σύντομα και συχνά περιέχουν συντομεύσεις όρων και συγκεκριμένη ορολογία ανά κατηγορία.

Label	# Train Docs	# Test Docs
acq	1,596	696
crude	253	121
earn	2,840	1,083
grain	41	10
interest	190	81
money-fx	206	87
ship	108	36
trade	251	75

Πίνακας 4.3: Κατανομή των κλασεων στο Reuters8

Κεφάλαιο 5

Πειραματική Αξιολόγηση

Οι τροποποιήσεις του μοντέλου γράφων λέξεων που παρουσιάστηκαν στο κεφάλαιο 3 αποτελούν επεκτάσεις του βασικού μοντέλου και μπορούν να εφαρμοστούν όλες μαζί ή ξεχωριστά η μία από την άλλη. Ωστόσο, αν εξεταστεί μόνο το πλήρες μοντέλο δεν θα είναι δυνατόν να προκύψει ποιες μέθοδοι βελτιώνουν πραγματικά την απόδοση ή ποιες δυσχεραίνουν τη λειτουργία του μοντέλου. Για να αξιολογήσουμε λοιπόν τις προτεινόμενες μεθόδους, αρχικά παρουσιάζεται το βασικό μοντέλο γράφων λέξεων και στη συνέχεια συγκρίνεται με κάθε νέα μέθοδο ξεχωριστά, οπότε οποιαδήποτε βελτίωση παρατηρείται θα μπορεί να πιστώνεται στη μέθοδο αυτή αποκλειστικά. Αρχικά θα παρουσιαστούν τα αποτελέσματα για το 20NG και στη συνέχεια για το R8. Θα δοθεί μεγαλύτερη βαρύτητα στα αποτελέσματα του 20NG, γιατί το R8 είναι αρκετά πιο εύκολο σύνολο για ταξινόμηση, οπότε οποιαδήποτε αλλαγή είναι δύσκολή.

5.1 Οργάνωση Πειραμάτων

Η οργάνωση των πειραμάτων είναι κοινή και για τις δύο συλλογές. Υπενθυμίζεται ότι στο 20NG έχουν αφαιρεθεί τα κείμενα που περιέχουν λιγότερο από 50 χαρακτήρες, γιατί πολλά από αυτά είναι σχεδόν κενά και περιέχουν πολλούς κενούς χαρακτήρες και διάφορα σύμβολα. Η διαδικασία που ακολουθήθηκε στα πειράματα είναι η εξής:

- Προεπεξεργασία κειμένων
- Κατασκευή γράφων λέξεων
- Σχηματισμός πίνακα βαρών TW-IDF
- Κατηγοριοποίηση κειμένων

Για την προεπεξεργασία, τα κείμενα διαχωρίστηκαν σε λεκτικές μονάδες και αναγνωρίστηκε το μέρος του λόγου για κάθε μια (tokenization και POS tagging), αφαιρέθηκαν οι αντωνυμίες και οι λέξεις που αναπαριστούν αριθμούς, έγινε μετατροπή σε μικρά γράμματα και εφαρμόστηκε lemmatization, για να πάρουμε τους τελικούς όρους που θα χρησιμοποιηθούν ως είσοδοι στο μοντέλο γράφων λέξεων. Όσο αφορά τις προτεινόμενες τροποποιήσεις, το coreference resolution προηγείται όλων αυτών διαδικασιών, ενώ η αντικατάσταση των collocations γίνεται στο τέλος.

Για τη κατασκευή των γράφων λέξεων χρειάζεται να προσδιοριστούν οι παράμετροι του μοντέλου, δηλαδή αν ο γράφος θα είναι κατευθυνόμενος ή μη, με βάρος ή χωρίς και το μέγεθος του παραθύρου που θα διατρέχει τους όρους. Οι γράφοι επιτρέπεται να περιέχουν βρόχους (self-loops), επειδή δεν παρατηρήθηκε κάποια σημαντική διαφορά στην απόδοση αν αφαιρεθούν. Στη περίπτωση που χρησιμοποιούνται word embeddings για τα βάρη, αυτά θα πρέπει να έχουν ήδη προπονηθεί σε κάποια συλλογή κειμένων. Επιπλέον, κατά τη κατασκευή των γράφων μπορούν να χρησιμοποιηθούν οι μέθοδοι μεταβλητού παραθύρου και ενίσχυσης κόμβων.

Στη συνέχεια, οι γράφοι χρησιμοποιούνται για να προκύψει το TW-IDF, ο πίνακας δηλαδή που περιέχει τα βάρη των όρων για όλα τα κείμενα. Η παράμετρος b στον τύπο του TW-IDF διατηρήθηκε σταθερή και ίση με 0.003, όπως προτείνεται στο [1]. Στα πειράματα που ακολουθούν, το βάρος κάθε όρου προκύπτει από το all-degree του αντίστοιχου κόμβου. Το in-degree, αν και δίνει καλύτερα αποτελέσματα για μικρά μεγέθη παραθύρου, όταν το παράθυρο μεγαλώνει η απόδοση πέφτει απότομα. Αντίθετα, το all-degree φαίνεται να επωφελείται από όλο και μεγαλύτερα μεγέθη παραθύρου και γι' αυτό προτιμήθηκε. Εξαιτίας αυτής της επιλογής, σε περίπτωση που χρησιμοποιείται γράφος με βάρη τα αποτελέσματα είναι τα ίδια, άσχετα με το αν ο γράφος είναι κατευθυνόμενος ή μη. Για το λόγο αυτό, στα διαγράμματα της επόμενη ενότητας, δεν διαχωρίζουμε τους γράφους με βάρος σε κατευθυνόμενους ή μη.
Για την κατηγοριοποίηση των κειμένων χρησιμοποιείται ο ταξινομητής Linear SVM από τη βιβλιοθήκη scikit-learn και ως μέτρο αξιολόγησης χρησιμοποιείται το macro F_1 score στο 20NG και το micro F_1 score στο R8. Για την ταξινόμηση, στον πίνακα TW-IDF έγινε L_2 κανονικοποίηση κατά γραμμή, ώστε οι τιμές που περιέχει να έρθουν στο κατάλληλο εύρος, για να μπορεί να συγκλίνει ο ταξινομητής. Τέλος, οι διάφορες επιλογές που έγιναν για τις διάφορες παραμέτρους των μεθόδων αναφέρονται στις αντίστοιχες ενότητες.

5.2 Αποτελέσματα στο 20 Newsgroups

Βασικό Μοντέλο Γράφων Λέξεων

Αρχικά, θα εξεταστεί το βασικό μοντέλο γράφων λέξεων για όλες τις δυνατές τροποποιήσεις και για διάφορα μεγέθη παραθύρου. Τα αποτελέσματα φαίνονται στις Εικόνες 5.1-5.3. Παρατηρούμε ότι:

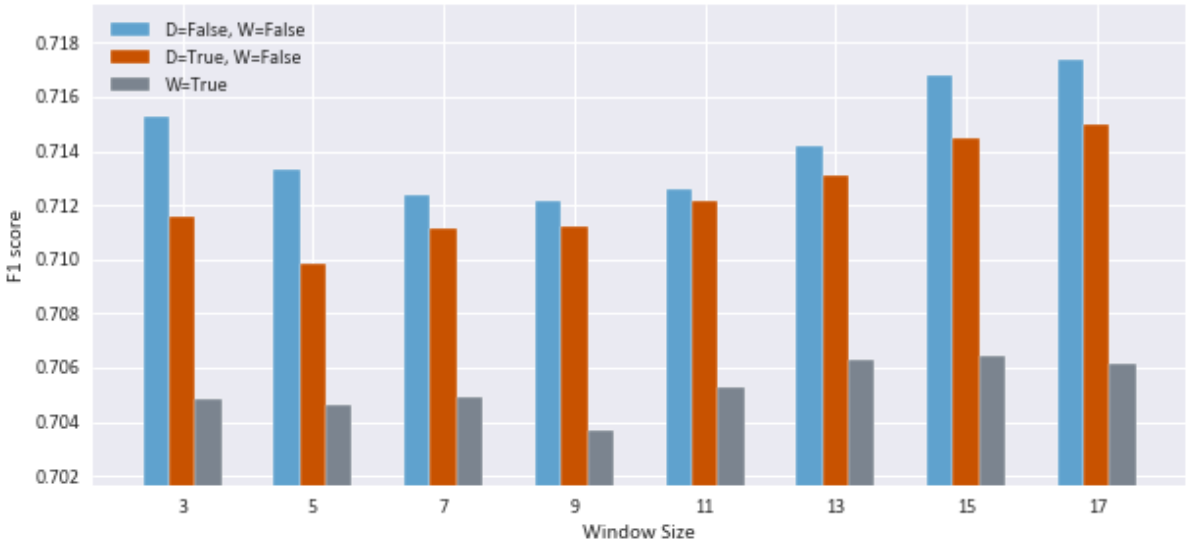
- Το μοντέλο γράφων λέξεων αποδίδει καλύτερα σε σχέση με το παραδοσιακό TF-IDF. Η μέγιστη απόδοση για το TW-IDF έφτασε το 0.7219 για μέγεθος παραθύρου 400, ενώ το TF-IDF έχει macro F_1 score ίσο με 0.7087.
- Το μέγεθος παραθύρου είναι καθοριστικός παράγοντας για την απόδοση. Αρχικά, για πολύ μικρό μήκος παραθύρου η απόδοση είναι αρκετά καλή, ενώ καθώς το παράθυρο μεγαλώνει η απόδοση παραμένει σταθέρη ή παρουσιάζει σταδιακή πτώση. Ωστόσο, για μήκος παραθύρου 11 και άνω φαίνεται μια ανοδική πορεία για το F_1 macro και τελικά φτάνει και ξεπερνάει το επίπεδο απόδοσης των μικρών παραθύρων.
- Τα καλύτερα αποτελέσματα τα παίρνουμε για πολύ μεγάλα παράθυρα, όπως φαίνεται στην Εικόνα 5.3. Το αποτέλεσμα αυτό δεν είναι αναμενόμενο καθώς είχαμε υποθέσει ότι το εύρος επιρροής των λέξεων δεν είναι πολύ μεγάλο, οπότε δεν θα έπρεπε να επωφελούμαστε από το μεγαλύτερο παράθυρο.
- Σε αυτό το σύνολο δεδομένων, οι γράφοι με βάρος δίνουν πάντα τα χειρότερα αποτελέσματα.
- Για μικρά παράθυρα, η μη κατευθυνόμενες ακμές είναι γενικά προτιμότερες, ενώ το αντίθετο ισχύει για μεγαλύτερα παράθυρα. Σε κάθε περίπτωση η διαφορά δεν είναι τόσο μεγάλη ώστε να επιβάλλει τη χρησιμοποίηση ή όχι κατευθυνόμενων ακμών.

Το κυριότερο συμπέρασμα που προκύπτει από τα παραπάνω αποτελέσματα είναι ότι τα μεγάλα παράθυρα βοηθούν πολύ στη βελτίωση της απόδοσης, κάτι που φυσικά δεν είναι

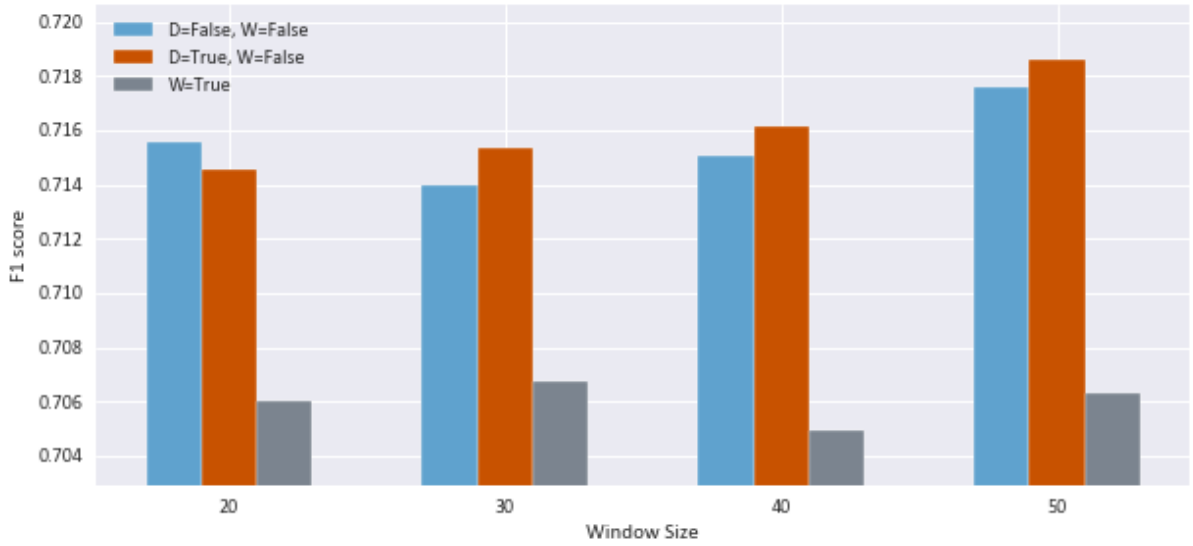
ιδανικό από άποψη πολυπλοκότητας, όπως φαίνεται και στον πίνακα 5.1, ούτε και αναμενόμενο, όπως αναφέρθηκε προηγουμένως. Οι μεγαλύτεροι γράφοι χρειάζονται περισσότερο χρόνο για να κατασκευαστούν και απαιτούν περισσότερη μνήμη. Η προφανής διαφορά του μεγαλύτερου παραθύρου είναι ότι δημιουργεί περισσότερες συνδέσεις και αυξάνει την περιοχή επιρροής των λέξεων, ωστόσο θεωρούμε ότι η βελτίωση οφείλεται στο πρόβλημα των γράφων λέξεων που παρουσιάζεται στην ενότητα 3.4, όπου το μοντέλο αδικεί τους όρους στις άκρες των κειμένων. Ο λόγος που το μεγαλύτερο μέγεθος παραθύρου βοηθάει είναι ο εξής: αν θεωρήσουμε ένα κείμενο με ξεχωριστούς όρους και μήκος $|d|$ και τον αντίστοιχο γράφο λέξεων με μέγεθος παραθύρου $\geq |d|$, τότε όλοι οι όροι θα συνδέονται με όλους και θα έχουν degree ίσο με $|d| - 1$. Επομένως, σε αυτή την περίπτωση το μέγεθος του κειμένου καθορίζει το μήκος του κειμένου ως ένα κατώτατο όριο για το βαθμό που μπορεί να λάβουν οι κόμβοι. Οι όροι που βρίσκονται στα άκρα των κειμένων δεν αδικούνται πλέον, γιατί το εύρος τους είναι πρακτικά ίδιο με όλους τους άλλους όρους.

Μέγεθος παραθύρου	3	10	17	50	100
Χρόνος κατασκευής	7.28 s	15.4 s	25.4 s	54.6 s	1min 31s

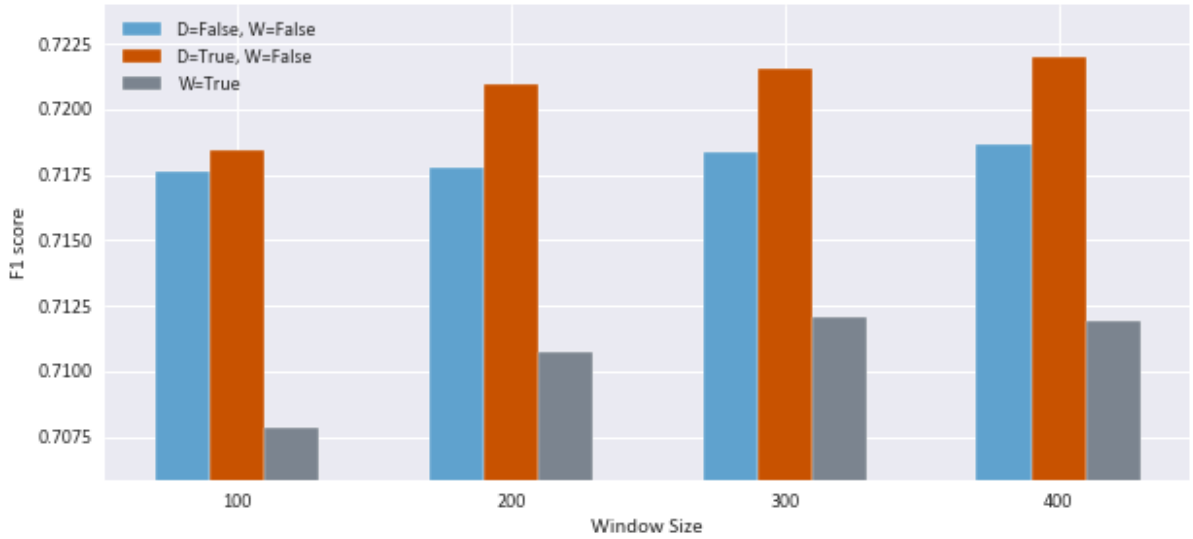
Πίνακας 5.1: Μέσος Χρόνος κατασκευής γράφων λέξεων για το σύνολο εκπαίδευσης 20 NG



Εικόνα 5.1: Βασικό μοντέλο για μήκη παραθύρου 3-17



Εικόνα 5.2: Βασικό μοντέλο για μήκη παραθύρου 20-50



Εικόνα 5.3: Βασικό μοντέλο για μήκη παραθύρου 100-400

Coreference Resolution

Όπως αναφέρθηκε στο Κεφάλαιο 3 επιλέξαμε να κρατήσουμε μόνο τα απλά coreferences που ξεκινούν από spans N λέξεων και καταλήγουν σε spans M λέξεων, όπου N,M μικροί φυσικοί αριθμοί και N≤M. Οι αναφορές που ξεκινούν μόνο από αντωνυμίες και ουσιαστικά δεν είναι τόσες πολλές στη συγκεκριμένη περίπτωση ώστε να υπάρχει αισθητή διαφορά στην απόδοση και γι' αυτό δεν περιοριστήκαμε σε αυτές. Στον πίνακα 5.2 φαίνονται πόσες αναφορές αναλύθηκαν για διάφορες τιμές των N και M.

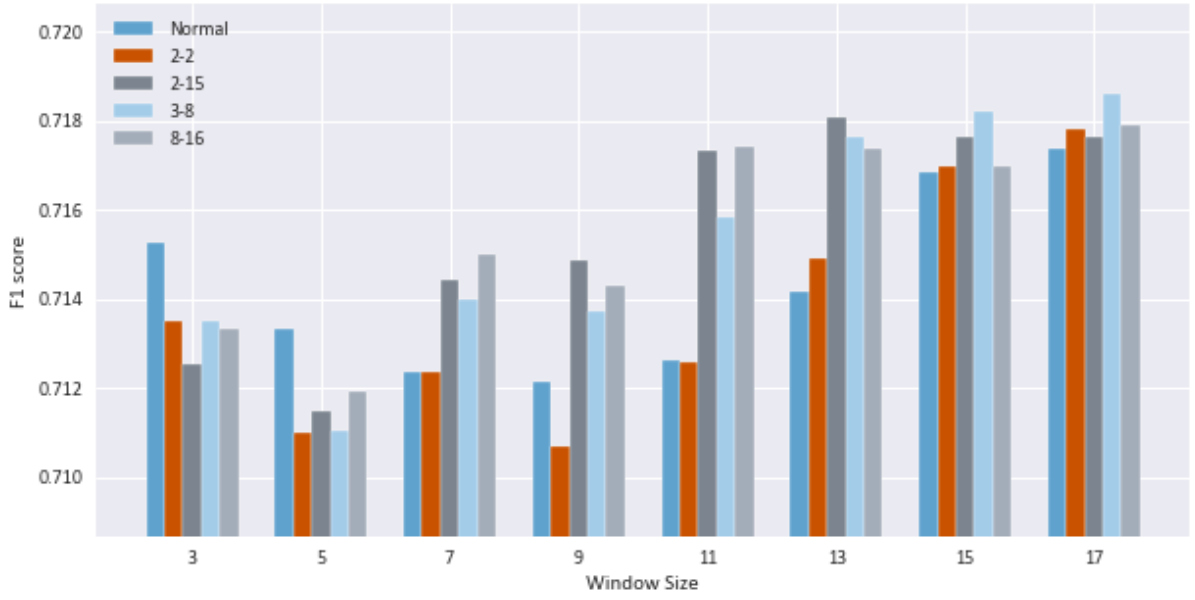
Είναι προφανές ότι τα περισσότερα coreferences ξεκινούν από μικρά spans και συνήθως καταλήγουν σε spans ίδιου μεγέθους, αφού το 93% των αναφορών που βρέθηκαν προκύπτουν για N≤2. Επιπλέον, παρατηρούμε ότι οι αναφορές προς μικρότερα spans είναι σχετικά λίγες κάτι που είναι αναμενόμενο και λογικό. Για παράδειγμα, υπάρχουν 33 αναφορές 4 σε 1 και 4770 αναφορές 1 σε 4. Ένα άλλο χρήσιμο συμπέρασμα που προκύπτει

είναι ότι άμα θέλουμε να συμπεριλάβουμε περισσότερες αναφορές πρέπει να αυξήσουμε το M και όχι το N, αφού για N≥4 τα coreferences είναι σχετικά λίγα.

N \ M	1	2	3	4	5	6	7	8	9	10	>10	
1	40460	19154	8298	4770	2398	1779	1338	904	731	557	2671	**83060**
2	1449	10593	2539	1010	507	413	210	205	135	107	667	**17835**
3	246	1071	2780	415	145	95	55	36	41	20	147	**5051**
4	33	182	236	691	83	39	29	16	4	7	33	**1353**
5	15	40	52	49	214	16	9	3	6	0	22	**426**
6	4	22	14	15	14	49	1	5	2	0	10	**136**
7	4	6	8	4	6	11	33	2	1	2	1	**78**
9	7	5	9	6	0	5	2	11	2	0	2	**49**
10	7	5	2	4	1	2	1	1	7	0	2	**32**
>10	4	2	3	1	2	0	1	0	0	6	4	**23**
	42229	**31080**	**13941**	**6965**	**3370**	**2409**	**1679**	**1183**	**929**	**699**	**3559**	**108043**

Πίνακας 5.2: Πλήθος N-M coreferences

Για λόγους απλότητας, τα αποτελέσματα παρουσιάζονται μόνο για μη κατευθυνόμενους γράφους χωρίς βάρη. Τα σχήματα N-M που εξετάστηκαν είναι τα 2-2, 2-15, 3-8, 8-16 όπου και πραγματοποιήθηκαν 71.656, 99.431, 100.870 και 106.606 αντικαταστάσεις αντίστοιχα.



Εικόνα 5.4: Coreference resolution για διάφορα σχήματα αναφορών N-M σε μη κατευθυνόμενους γράφους χωρίς βάρος

Από τα αποτελέσματα προκύπτει ότι:

- Το coreference resolution μπορεί να βοηθήσει στη βελτίωση της απόδοσης. Η αντικατάσταση μόνο των 2-2 αναφορών δεν δίνει σταθερά αποτελέσματα, αλλά στις υπόλοιπες περιπτώσεις η απόδοση για μήκη παραθύρου 7 και πάνω παρουσιάζει βελτίωση. Η σημαντικότερη διαφορά παρατηρείται στο σημείο που το βασικό μοντέλο κάνει κοιλία, από μέγεθος παραθύρου 7 έως 13.

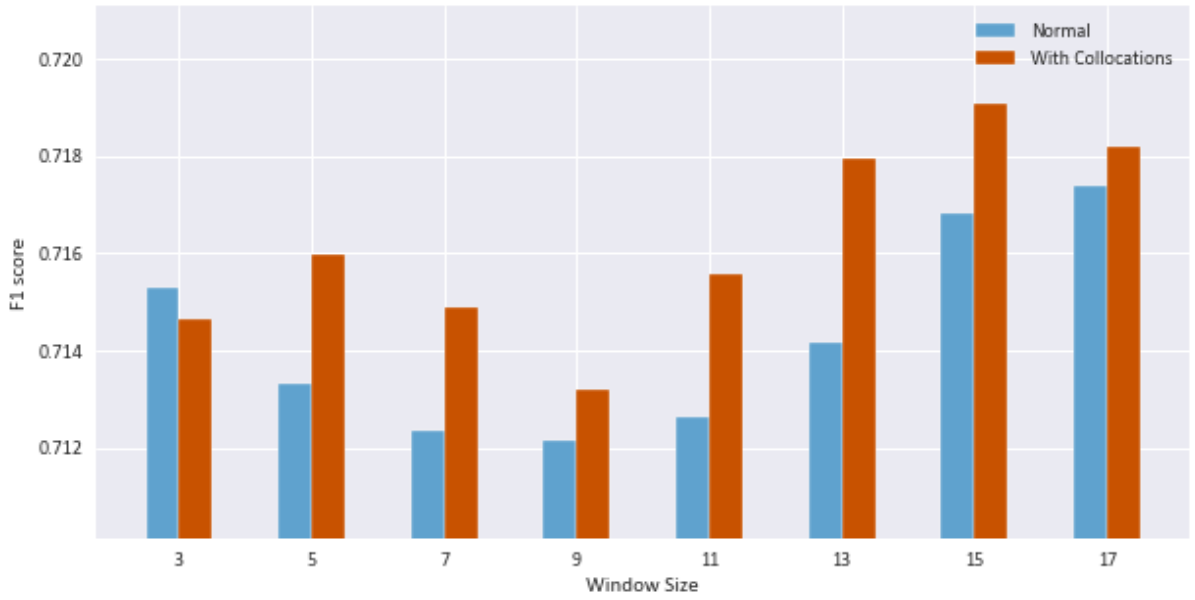
- Σε κάθε περίπτωση, για μικρά μεγέθη παραθύρου η απόδοση είναι χειρότερη από το βασικό μοντέλο. Το μικρό παράθυρο πιθανώς δεν έχει τη δυνατότητα να συνδέσει επαρκώς τους αναφερόμενους όρους με το υπόλοιπο κείμενο.
- Η αύξηση του M βοηθάει γενικά όπως φαίνεται από τις περιπτώσεις 2-2 και 2-15. Το μεγαλύτερο M εισάγει περισσότερους όρους σε διαφορετικά σημεία του κειμένου, με αποτελέσμα να δημιουργούνται πολλές νέες συνδέσεις. Επομένως, ο στόχος του coreference resolution να δώσει περισσότερο βάρος στους αναφερόμενους όρους μέσω καινούργιων συνδέσεων επιτεύχθηκε.

Collocation Detection

Για την εξαγωγή των collocations πρέπει να καθοριστούν δύο παράμετροι:

- min count, που είναι το ελάχιστο πλήθος φορών που πρέπει να εμφανίζεται το collocation μέσα στη συλλογή κειμένων για να θεωρηθεί έγκυρο.
- threshold, το όριο δηλαδή που πρέπει να περάσει η τιμή ενός collocation για να θεωρηθεί έγκυρο.

Στα πειράματα κρατήσαμε το $min\ count$ σταθερό και ίσο με 5 και εξετάστηκαν διάφορες τιμές για το $threshold$ στο εύρος 10-2000. Η διαδικασία για την αναγνώριση collocation έγινε συνολικά δύο φορές, ώστε να προκύψουν και μεγαλύτερες εκφράσεις. Επιπλέον, προτιμήθηκε το απλό μοντέλο collocation, όπου η τιμή του δεν είναι κανονικοποιημένη, γιατί φάνηκε να δίνει καλύτερα αποτελέσματα, ενώ ο γράφος που χρησιμοποιήθηκε είναι μη κατευθυνόμενος χωρίς βαρος. Στην παρακάτω Εικόνα παρουσιάζονται τα αποτελέσματα της πειραματικής αξιολόγησης για threshold ίσο με 70:

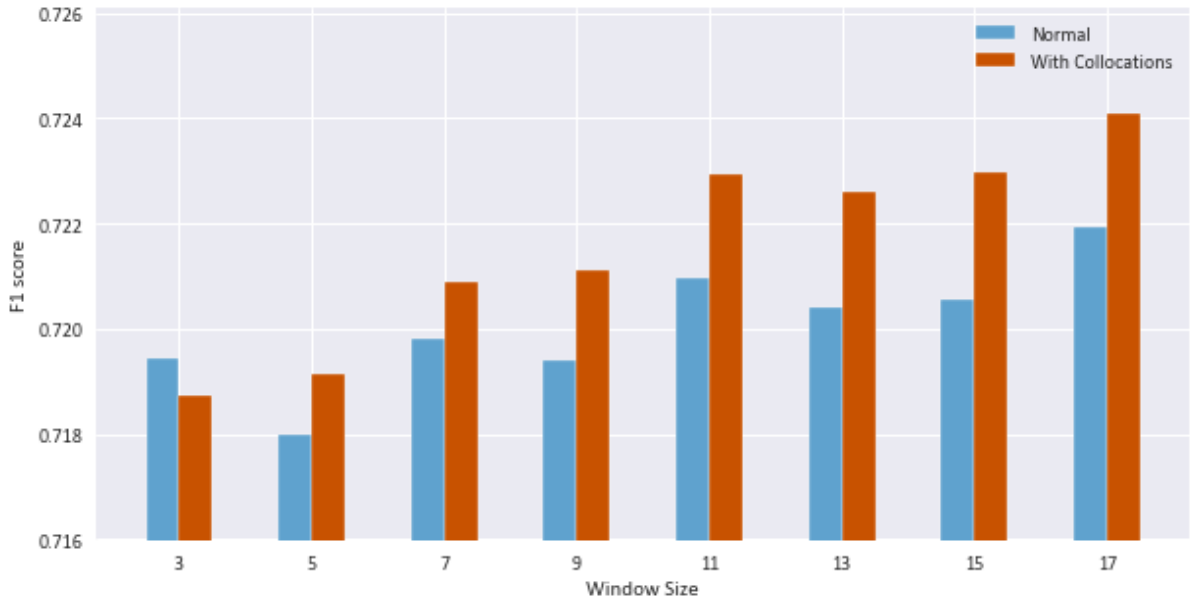


Εικόνα 5.5: Collocation detection για min count = 5 και threshold = 70

Τα σχόλια που προκύπτουν είναι:

- Το collocation detection σαφώς και βοηθάει στην βελτίωση της απόδοσης. Για threshold ίσο με 70 παρατηρούμε αύξηση της απόδοσης για κάθε μέγεθος παραθύρου εκτός από 3 που κυμαίνεται από 0.01 έως και 0.04. Στη περίπτωση αυτή εντοπίστηκαν 1855 collocations την πρώτη φορά και 1749 τη δεύτερη και έγιναν συνολικά 65349 αντικαταστάσεις. Επιπλέον οι αλλαγές έγιναν σε 6218 και 3630 train και test κείμενα αντίστοιχα.
- Μερικά από τα collocations που αναγνωρίστηκαν είναι γενικά και αντιστοιχούν σε πραγματικές οντότητες όπως το las_vegas, baltimore_skipjacks, saudi_arabia, yasser_arafat, pizza_hut και άλλα. Αντίθετα υπάρχουν και αρκετά collocations τα οποία είναι ειδικά σε αυτή τη συλλογή κειμένων. όπως το bake_timmons, feustel_n9myi, samuel_shahmuradian, 0:00 utc, όπου πρόκειται κυρίως για ονόματα χρηστών ή για κάποιες εκφράσεις που χρησιμοποιούσαν ειδικά οι χρήστες σε αυτό το newsgroup. Θεωρούμε ότι και τα δύο είδη είναι χρήσιμα και για αυτό δεν χρησιμοποιήθηκε και κάποιο άλλο σύνολο δεδομένων για το collocation detection, αφού έτσι θα χάναμε τις ειδικές εκφράσεις που εμφανίζονται μόνο στο 20NG.
- Για τον καθορισμό του threshold πρέπει να είμαστε ιδιαίτερα αυστηροί, αφού μόνο στο διάστημα 10-110 παρατηρήθηκαν τιμές για τις οποίες υπάρχει αισθητή και σταθερή διαφορά.

Ακολούθως, εξετάστηκε το κατά πόσο η μέθοδος collocation detection μπορεί να συνδυαστεί με τη μέθοδο Rebase, η οποία θα εξεταστεί αργότερα, και κατά πόσο το *threshold* είναι ευαίσθητο σε αλλαγές. Πράγματι, σε αυτήν την περίπτωση το διάστημα της παραμέτρου $threshold$ για το οποίο παρατηρείται βελτίωση στην απόδοση κυμαίνεται από 500 έως και 2000 και είναι σημαντικά μεγαλύτερο από πριν. Η παρακάτω Εικόνα προκύπτει για threshold ίσο με 700, όπου πάλι παρατηρούμε ανάλογη βελτίωση με πριν.



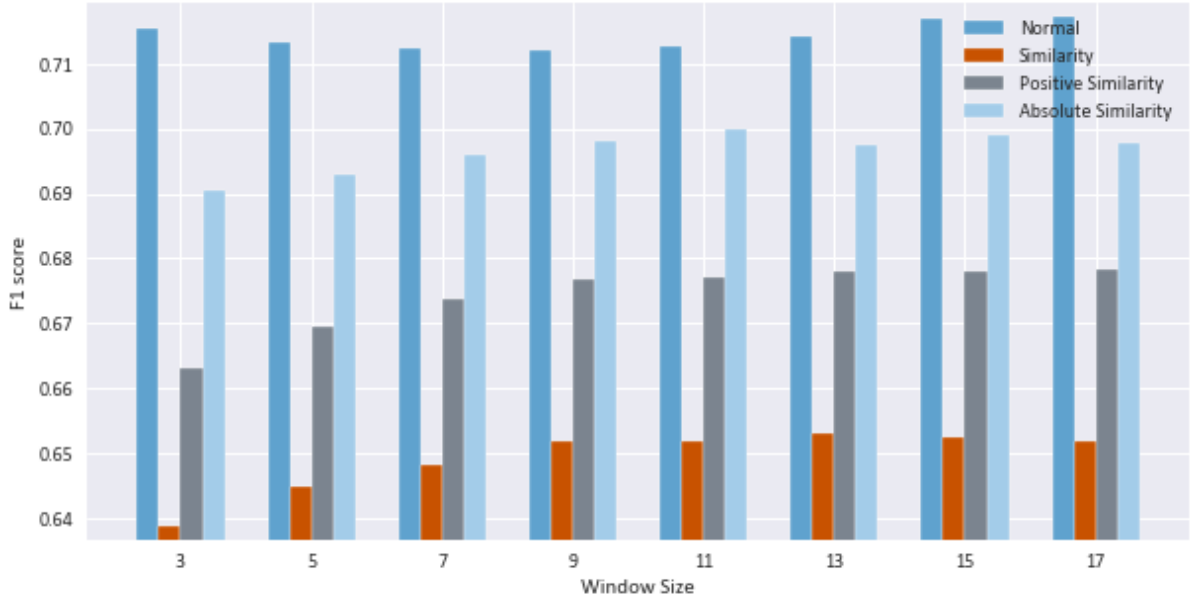
Εικόνα 5.6: Collocation detection μαζί με Rebase για min count = 5 και threshold = 700

Word Embeddings

Το γενικό σχήμα ανάθεσης βάρους στις ακμές με χρήση των word embeddings που χρησιμοποιήθηκε είναι το εξής:

$$w_{i,j} = a + b * f(sim(i,j))$$

Αρχικά, εξετάστηκε αν η απόσταση ομοιότητας αποτελεί από μόνη της αντιπροσωπευτικό βάρος για τις ακμές. Τα αποτελέσματα που προκύπτουν για $a = 0, b = 1$ και $f(x) = identity(x)$, $f(x) = \max(x, 0)$ και $f(x) = abs(x)$ παρουσιάζονται στο σχήμα 5.7. Παρατηρούμε ότι αν απλά χρησιμοποιήσουμε την απόσταση ομοιότητας, η απόδοση παρουσιάζει μεγάλη πτώση, ενώ αν κρατήσουμε μόνο τις θετικές αποστάσεις η κατάσταση βελτιώνεται λίγο. Η χρήση της απόλυτης τιμής δίνει τα καλύτερα αποτελέσματα μεταξύ των νέων μοντέλων, αλλά αποτυγχάνει να ξεπεράσει το παραδοσιακό μοντέλο γράφων λέξεων και παρουσιάζει μείωση του F_1 score κατά 0.1 έως 0.2. Από την απόδοση των τριών νέων μοντέλων, μπορούμε να συμπεράνουμε ότι για την ανάθεση βάρους στις ακμές δεν παίζει ρόλο η ομοιότητα των όρων μεταξύ τους, αλλά η συνύπαρξή τους μέσα σε ένα παράθυρο και άρα η σημαντικότητα ενός όρου δεν καθορίζεται από το πόσο πολύ μοιάζει με τις κοντινές του λέξεις. Η εκδοχή για $f(x) = abs(x)$ αποδίδει συγκριτικά καλύτερα, γιατί η συνάρτηση απόλυτης τιμής συγκρατεί καλύτερα το γεγονός της συνύπαρξης, είτε αυτή αφορά όρους που μοιάζουν σημασιολογικά είτε όχι. Από την άλλη πλευρά, η συνάρτηση $f(x) = identity(x)$ τιμωρεί τους όρους που είναι αντίθετοι των κοντινών τους, ακόμα και αν αυτοί υπάρχουν πολλές φορές μέσα στο κείμενο και θα είχαν μεγάλο βάρος στο απλό μοντέλο γράφων λέξεων. Επιπλέον, αν κάποιο όροι δεν έχουν καμία ομοιότητα, τότε η πληροφορία της συνύπαρξης τους θα χαθεί σε όλα τα νέα μοντέλα και για αυτό παρατηρείται αυτή η πτώση στην απόδοση.



Εικόνα 5.7: Απόσταση ομοιότητας word embeddings ως βάρη των ακμών με χρήση των συναρτήσεων identity(x), max(x,0) και abs(x). Μη κατευθυνόμενος γράφος χωρίς βάρη

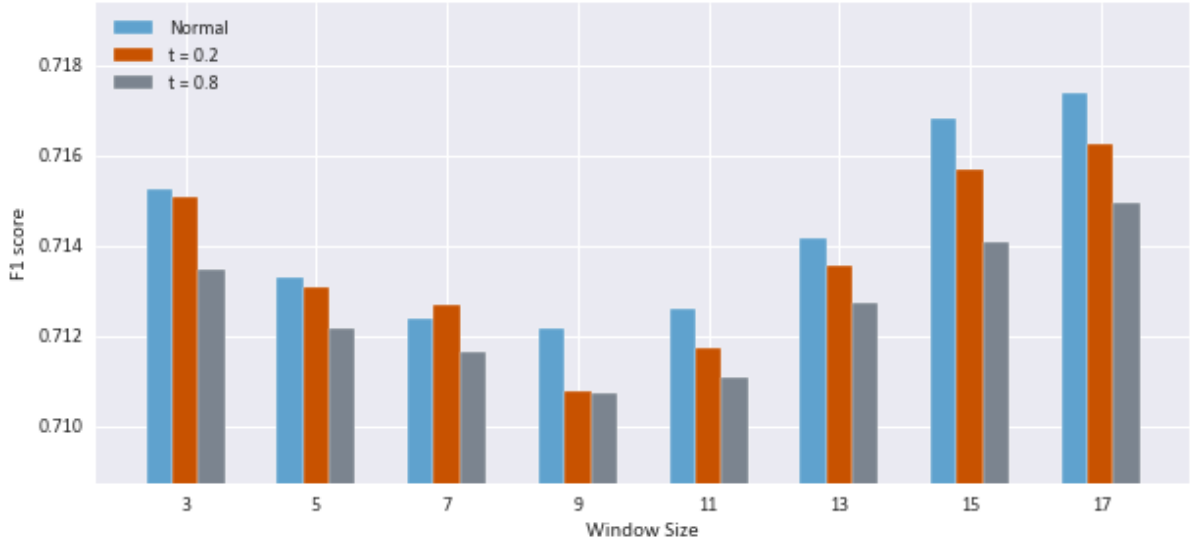
Το ερώτημα που προκύπτει τώρα είναι αν είναι εφικτό να συνδυαστεί η πληροφορία της ομοιότητας με αυτή της συνύπαρξης. Για το σκοπό αυτό εξετάστηκαν δύο διαφορετικές εκδοχές ανάθεσης βαρών:

$$w_{i,j} = 1 + t * abs(sim(emb_i, emb_j))$$

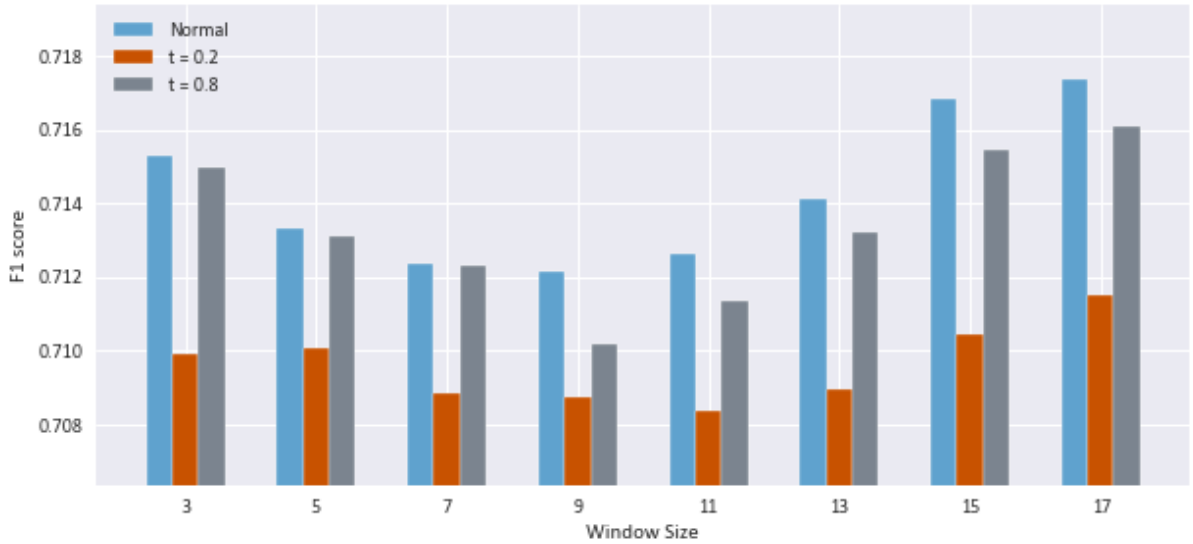
και

$$w_{i,j} = t + (1 - t) * abs(sim(emb_i, emb_j))$$

Η διαφορά των δύο εκδοχών είναι το εύρος τιμών τους βάρος. Η πρώτη εκδοχή έχει ελάχιστη τιμή το 1 και η συνεισφορά της συνύπαρξης είναι σταθερή, ενώ στη δεύτερη η μέγιστη τιμή είναι 1 και οι συνεισφορές είναι αντιστρόφως ανάλογες. Σε κάθε περίπτωση, όσο μεγαλύτερη είναι η ομοιότητα, τόσο μεγαλύτερο είναι το βάρος. Η συνάρτηση $f(x) = abs(x)$ επιλέχθηκε για τους λόγους που αναλύθηκαν παραπάνω. Τα αποτελέσματα φαίνονται στις Εικόνες 5.8 και 5.9 για δύο διαφορετικές τιμές της παραμέτρου t.



Εικόνα 5.8: Σχήμα ανάθεσης βάρους $w_{i,j} = 1 + t * abs\left(sim\left(emb_i, emb_j\right)\right)$ για $t = 0.2$ και $t = 0.8$



Εικόνα 5.9: Σχήμα ανάθεσης βάρους $w_{i,j} = t + (1 - t) * abs\left(sim\left(emb_i, emb_j\right)\right)$ για $t = 0.2$ και $t = 0.8$

Είναι προφανές ότι δεν παρατηρείται κάποια σταθερή βελτίωση στην απόδοση για τις δύο νέες εκδοχές σε σχέση με το απλό μοντέλο γράφων λέξεων, ενώ μάλιστα η επιρροή της απόστασης ομοιότητας στην ανάθεση του βάρους, είναι αντιστρόφως ανάλογη με την απόδοση. Ειδικότερα, στην πρώτη εκδοχή, όσο πιο μικρή είναι η παράμετρος t που καθορίζει το βάρος της απόστασης, τόσο πιο καλή είναι η απόδοση. Αντίστοιχα, στη δεύτερη μέθοδο όταν το t αυξάνεται, τα αποτελέσματα βελτιώνονται, αλλά δεν φτάνουν την απόδοση το απλού μοντέλου.

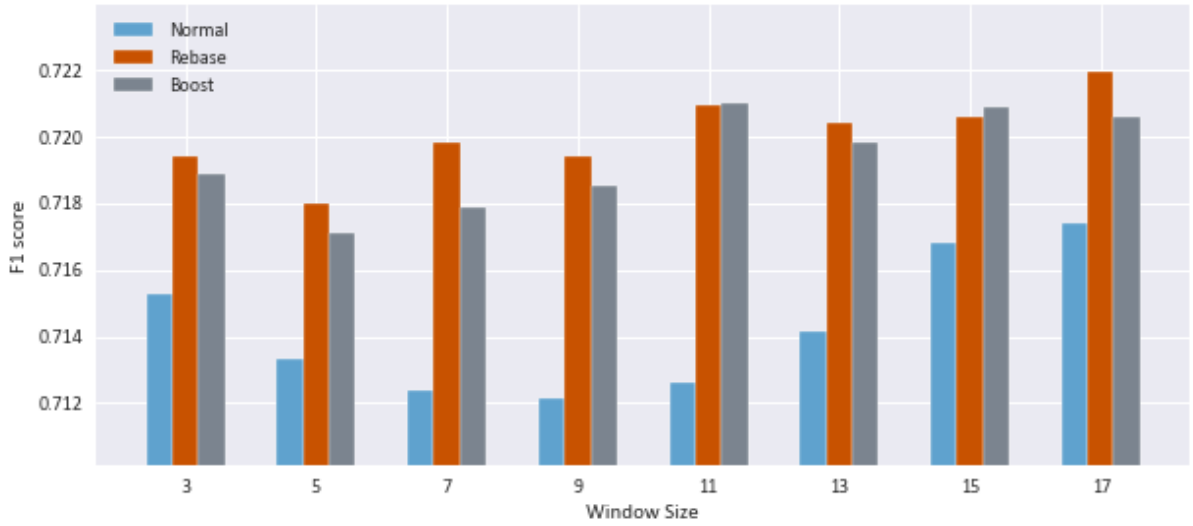
Παρόμοια αποτελέσματα παρατηρήθηκαν και στην περίπτωση όπου η επανάληψη της ακμής λαμβάνεται υπόψη για το βάρος των ακμών, οπότε κρίθηκε σκόπιμο να μην παρουσιαστούν.

Ενίσχυση Κόμβων

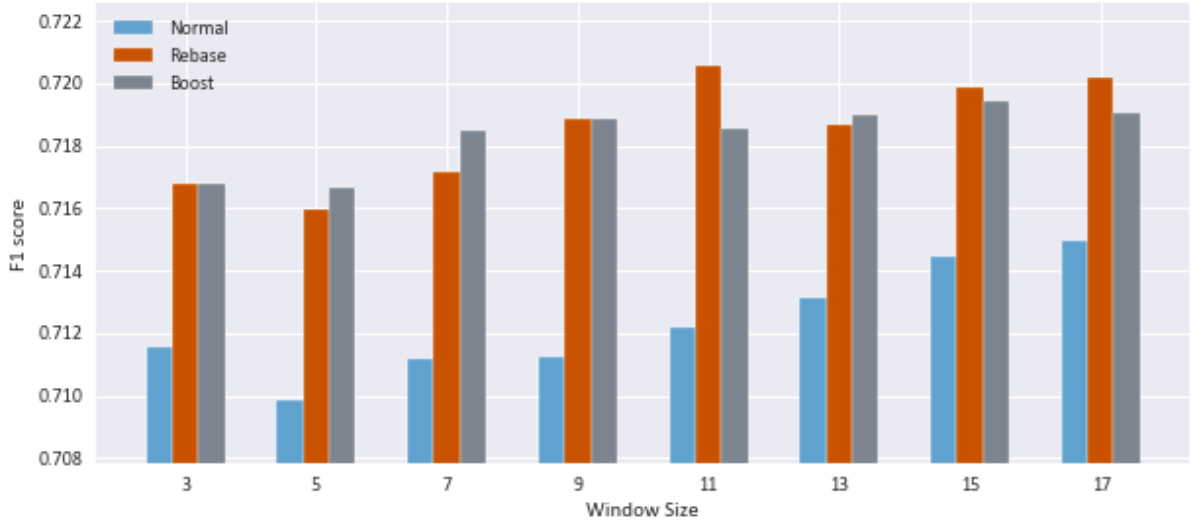
Οι δύο τεχνικές ενίσχυσης κόμβων του γράφου που θα εξεταστούν σε αυτή την ενότητα είναι το Rebase και το Boost. Η μόνη παράμετρος που πρέπει να οριστεί είναι το όριο των βαρών για την πρώτη μέθοδο, το οποίο όμως έχει καθοριστεί στο Κεφάλαιο 4 και δεν πραγματοποιήθηκε περαιτέρω αναζήτηση. Τα αποτελέσματα για όλες τις διαμόρφωσεις των κόμβων φαίνονται στις εικόνες 5.10-5.12. Συμπεραίνουμε τα εξής:

- Και οι δύο μέθοδοι ενίσχυσης των γράφων βοηθούν σημαντικά στην απόδοση για όλες τις διαμορφώσεις των κόμβων και για όλα τα μεγέθη παραθύρου. Στην Εικόνα 5.11 παρατηρούμε διαφορές που κυμαίνονται από 0.04 έως και 0.08 και η απόδοση φτάνει το 0.722 για μέγεθος παραθύρου 17. Η σημαντικότερη βελτίωση παρουσιάζεται στους γράφους με βάρη, όπου η μέθοδος Boost πετυχαίνει έως 0.1 καλύτερο F_1 score.
- Δεν υπάρχει αισθητή διαφορά μεταξύ των δύο μεθόδων, εκτός από όταν χρησιμοποιούμε γράφους με βάρη. Σε αυτήν την περίπτωση, το Boost πετυχαίνει σταθερά καλύτερη απόδοση από το Rebase. Αυτό οφείλεται στο γεγονός ότι το όριο που έχουμε θέσει δεν είναι κατάλληλο για γράφους με βάρη, γιατί αυτοί λαμβάνουν υπόψη τις επαναλήψεις ακμών, οπότε και το degree των κόμβων είναι μεγαλύτερο συγκριτικά με τους γράφους δίχως βάρη. Με άλλα λόγια, αν δεν μπορούμε να βρούμε ένα κατάλληλο όριο, η μέθοδος Boost είναι προτιμότερη, γιατί απλά ενισχύει τους όρους στις δύο άκρες των κειμένων, ανεξάρτητα από το degree των αντίστοιχων κόμβων.

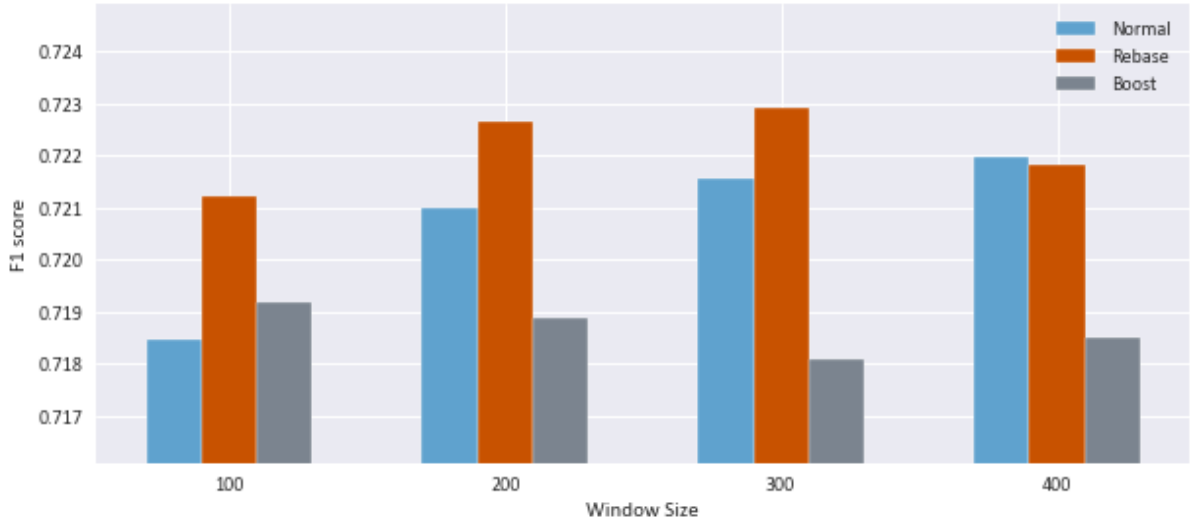
Επιπλέον, στην Εικόνα 5.13 φαίνεται ότι η βελτίωση που παρατηρήσαμε στα μικρά παράθυρα δεν μεταφέρεται και στα πολύ μεγάλα παράθυρα, ειδικά στην περίπτωση του Boost. Αυτό ενισχύει την υπόθεση που είχαμε κάνει, ότι δηλαδή η μεγάλη απόδοση των πολύ μεγάλων παράθυρων οφείλεται σε μεγάλο βαθμό στο πρόβλημα της άνισης αντιπροσώπευσης. Επομένως, το πλεονέκτημα των μεθόδων αυτών είναι ότι δεν υπάρχει πλέον ανάγκη να χρησιμοποιήσουμε πολύ μεγάλα παράθυρα που ανεβάζουν κατά πολύ τον χρόνο κατασκευής των γράφων, αλλά μπορούμε να περιοριστούμε και σε πιο μικρά, με παρόμοια απόδοση.



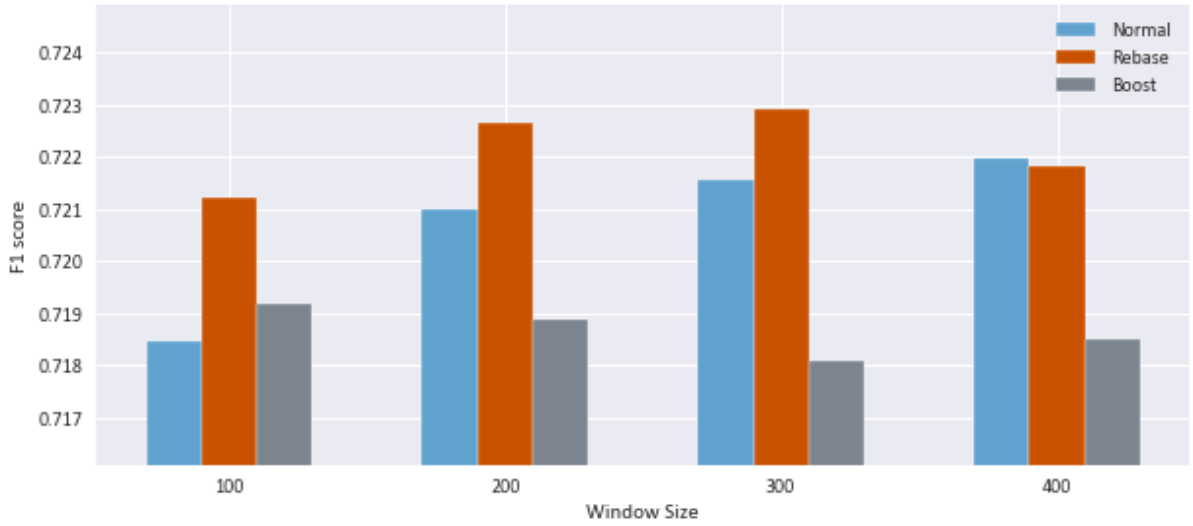
Εικόνα 5.10: Rebase και Boost για μη κατευθυνόμενους γράφους χωρίς βάρος



Εικόνα 5.11: Rebase και Boost για κατευθυνόμενους γράφους χωρίς βάρος



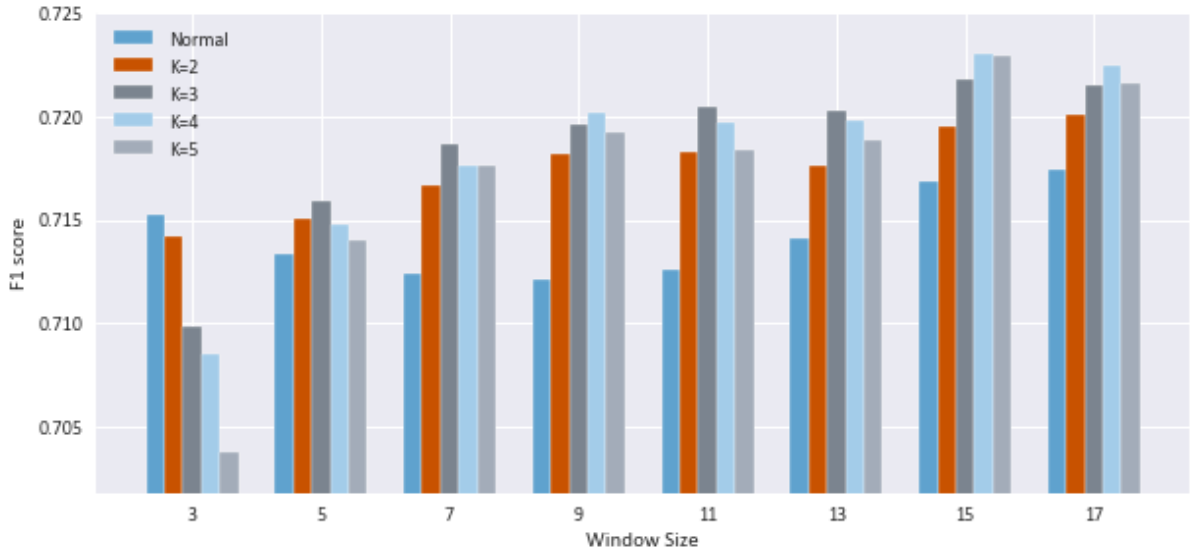
Εικόνα 5.12: Rebase και Boost για γράφους με βάρος



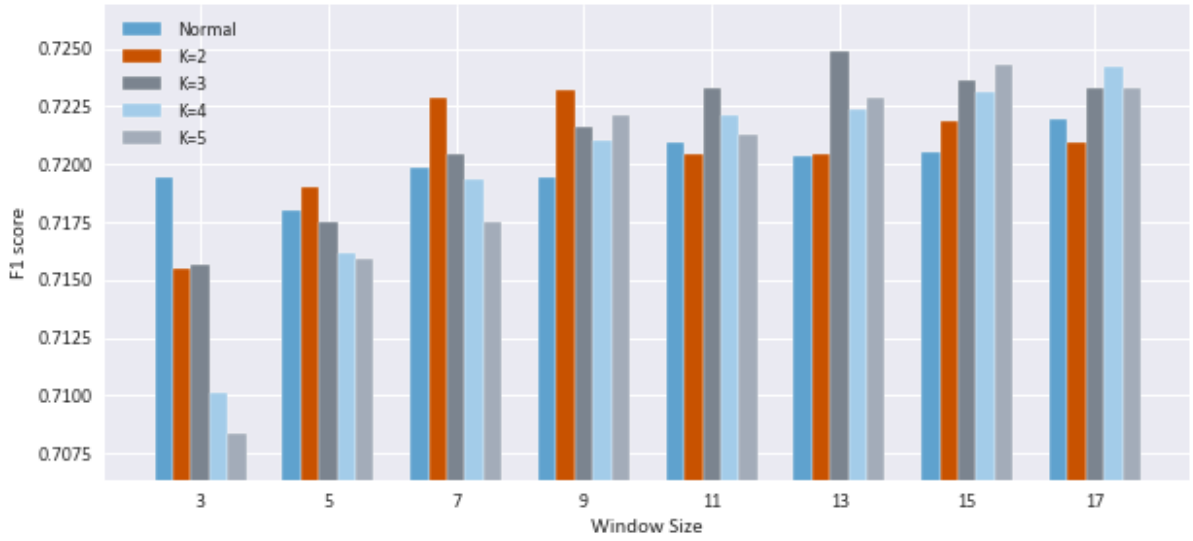
Εικόνα 5.13: Rebase και Boost για κατευθυνόμενους γράφους χωρίς βάρος και μήκος παραθύρου 100-400

Μεταβλητό Μέγεθος Παραθύρου

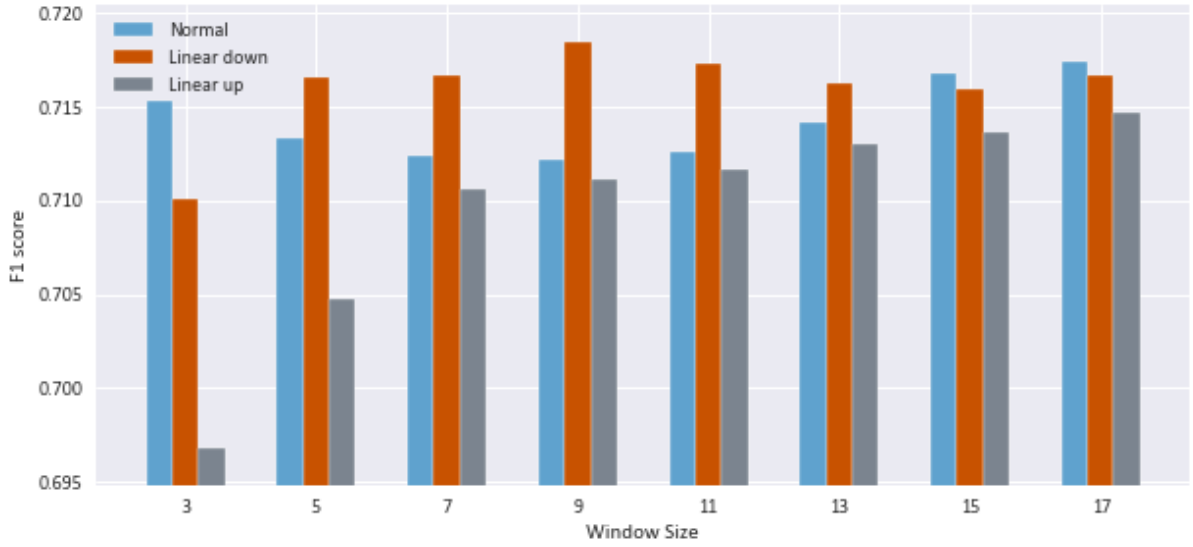
Για τον καθορισμό των μεγεθών των παραθύρων εξετάστηκαν διάφοροι μέθοδοι που βασίζονται μόνο στη σειρά των όρων μέσα στο κείμενο, ώστε και πάλι να δώσουμε μεγαλύτερη προσοχή στα άκρα των κειμένων ή και σε άλλες περιοχές. Για την παραγωγή των αποτελεσμάτων χρησιμοποιήθηκαν μη κατευθυνόμενοι γράφοι χωρίς βάρη. Στην Εικόνα 5.14 το μέγεθος παραθύρου (ws) στους πρώτους ws όρους έχει αλλάξει σε $K * ws$, για $K = 2, 3, 4, 5$, ενώ οι υπόλοιποι όροι έχουν μέγεθος παραθύρου ws. Η ίδια μέθοδος έχει χρησιμοποιηθεί και για τα αποτελέσματα στην Εικόνα 5.15, όπου όμως έχει χρησιμοποιηθεί και η μέθοδος Boost για ενίσχυση των κόμβων, για να καθοριστεί εάν οι τροποποιήσεις μπορούν να εφαρμοστούν μαζί. Επιπλέον, στην Εικόνα 5.16 έχει γίνει γραμμική μείωση από $4 * ws$ για τον πρώτο όρο μέχρι ws για τον τελευταίο και αντίστροφα αύξηση από ws μέχρι $4 * ws$.



Εικόνα 5.14: Μεταβολή μεγέθους παραθύρου στους πρώτους όρους με πολλαπλασιασμό



Εικόνα 5.15: Μεταβολή μεγέθους παραθύρου στους πρώτους όρους με πολλαπλασιασμό σε συνδυασμό με τη μέθοδο Rebase



Εικόνα 5.16: Γραμμική μεταβολή μήκους παραθύρου από 4*ws για το πρώτο όρο μέχρι ws για τον τελευταίο και αντίστροφα

Από τα αποτελέσματα μπορούμε να συμπεράνουμε ότι:

- Η μέθοδος μεταβλητού μεγέθους παραθύρου είναι πολύ χρήσιμη και μπορεί να δώσει σημαντικά καλύτερα αποτελέσματα στην ταξινόμηση των κειμένων. Στην Εικόνα 5.14 παρατηρούμε διαφορές στην απόδοση, όπου σε πολλές περιπτώσεις υπάρχει αύξηση κατά 0.25 και πάνω. Ωστόσο, για μικρό μέγεθος παραθύρου η απόδοση είναι χειρότερη ή η αύξηση είναι μικρή και αυτό οφείλεται στο γεγονός ότι οι πρώτοι όροι έχουν δυσανάλογα πολλές συνδέσεις σε σχέση με τους υπόλοιπους και κυριαρχούν σε υπερβολικό βαθμό.
- Στην Εικόνα 5.15 παρατηρούμε ότι η μέθοδος μεταβλητού παραθύρου μπορεί να συνδυαστεί με τη μέθοδο Boost και μάλιστα δίνει αρκετά καλύτερα αποτελέσματα απ΄ότι αν χρησιμοποιούσαμε μόνο τη μέθοδο Boost ή μόνο τη μέθοδο μεταβλητού παραθύρου. Με άλλα λόγια, η μέθοδος αυτή είναι χρήσιμη, όχι απλά ως ένα

διαφορετικό μέτρο για τη ενίσχυση των ακραίων κόμβων, αλλά και ως ένα γενικότερο μέτρο για την απόδοση καταλληλότερου βάρους στα διάφορα μέρη του κειμένου.

- Η γραμμική αυξομείωση του βάρους δεν δίνει καλύτερη απόδοση αλλά από την Εικόνα 5.16 προκύπτει ότι είναι προτιμότερο να έχουμε πιο μεγάλο παραθύρου στην αρχή παρά στο τέλος του κειμένου. Το μεγαλύτερο μέγεθος παραθύρου σημαίνει και μεγαλύτερη σημασία του κόμβου, άρα τουλάχιστον για αυτή τη συλλογή κειμένων, η αρχή των κειμένων είναι πιο σημαντική για την ταξινόμηση σε σχέση με το τέλος των κειμένων.

Ensembles Γράφων Λέξεων

Οι μέθοδοι soft vote και stacking χρησιμοποιούν τις πιθανότητες που αναθέτει το μοντέλο σε κάθε κλάση. Ωστόσο, ο ταξινομητής Linear SVM που χρησιμοποιούμε δεν δίνει ως έξοδο αυτές τις πιθανότητες, οπότε χρησιμοποιήθηκε η μέθοδος CalibratedClassifierCV της βιβλιοθήκης scikit-learn για να αποκτήσουμε αυτές τις πιθανότητες. Για το λόγο αυτό, η απόδοση των απλών μοντέλων είναι διαφορετική στις περιπτώσεις αυτές, αφού η μέθοδος χρησιμοποιεί cross validation, για να κάνει τη βαθμονόμηση των πιθανοτήτων (probability calibration).

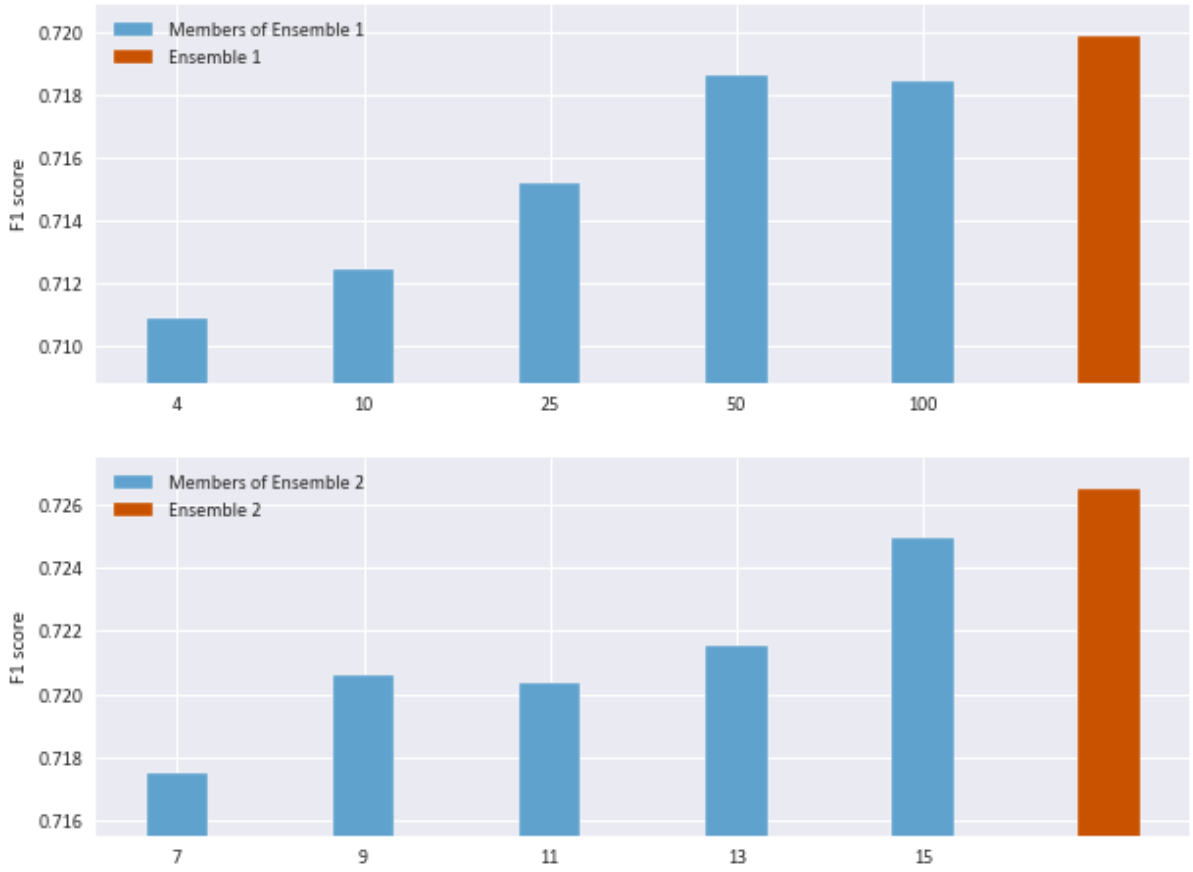
Καταρχήν, θα εξετάσουμε τις μεθόδους hard vote και soft vote, όπου για την αξιολόγηση τους χρησιμοποιήθηκαν διάφορα ensembles, τα χαρακτηριστικά των οποίων φαίνονται στον παρακάτω πίνακα:

	Γράφος	**Μέγεθος παραθύρου**	**Βαρύτητα**	**Μέθοδος**
Ensemble 1	Κατευθυνόμενος, χωρίς βάρη	4, 10, 25, 50, 100	1,1,1,2,3	-
Ensemble 2	Κατευθυνόμενος, χωρίς βάρη	7, 9, 11, 13, 15	1,1,1,2,3	Μεταβλητό παράθυρο
Ensemble 3	Κατευθυνόμενος, χωρίς βάρη	4, 10, 25, 50, 100	F_1 scores	-
Ensemble 4	Μη κατευθυνόμενος, χωρίς βάρη	11, 12, 13	4, 1, 2	Rebase

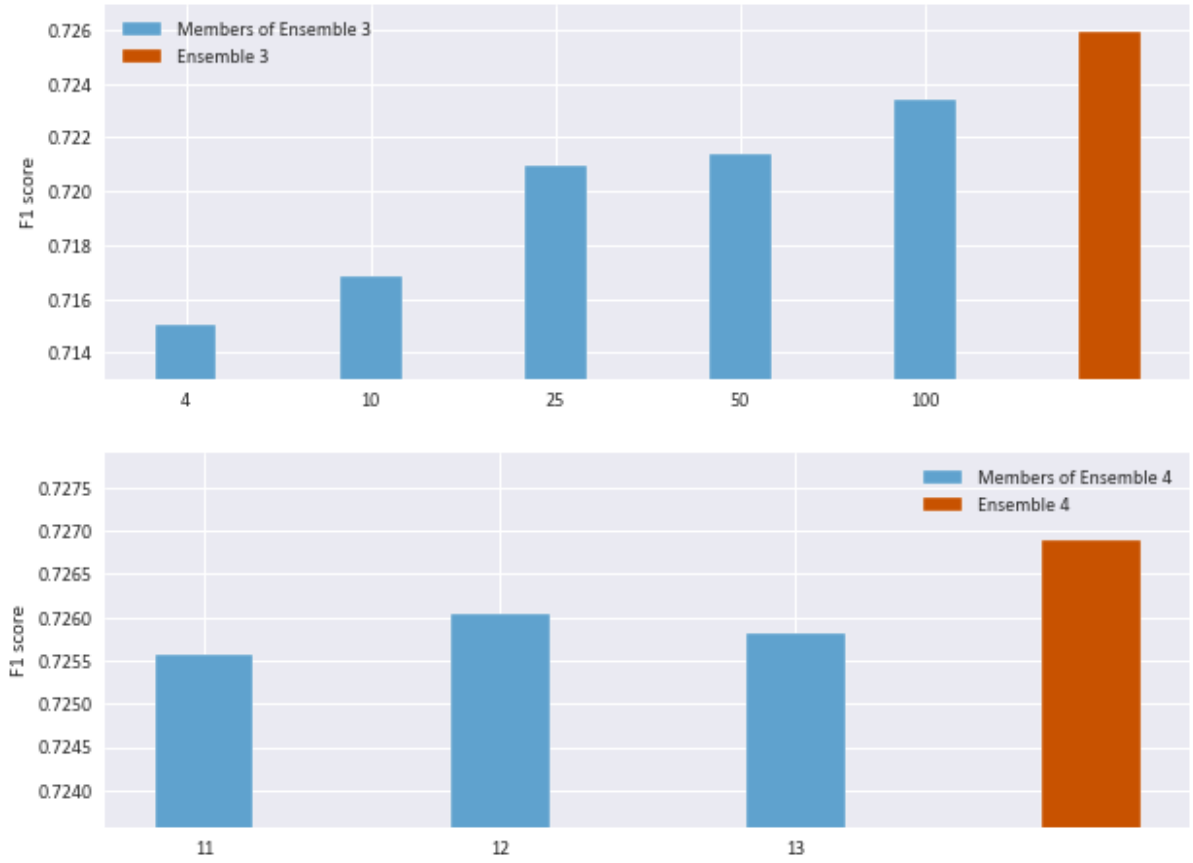
Πίνακας 5.3: Ensembles για hard και soft vote στο 20NG

Παρατηρούμε τα εξής:

- Ακόμα και η χρήση απλών μεθόδων ensembles, όπως η ψηφοφορία, μπορεί να είναι χρήσιμη αφού προκύπτει βελτίωση του F_1 της τάξης 0.01 έως 0.02 σε σχέση με το καλύτερο μέλος της ομάδας.
- Η αναζήτηση των ensembles και της αντίστοιχης βαρύτητας των μελών δεν είναι εύκολη, καθώς χρειαζόταν προσεκτική επιλογή τόσο των απλών μοντέλων όσο και των βαρών που θα έχουν αυτά. Τα voting ensembles φαίνεται να δουλεύουν καλύτερα όταν υπάρχουν ένα ή δύο πολύ καλά μοντέλα μέσα στη ομάδα και τα οποία θα πρέπει φυσικά να έχουν και μεγαλύτερος βάρος.



Εικόνα 5.17: Hard Vote Ensemble για Ensemble 1 και 2

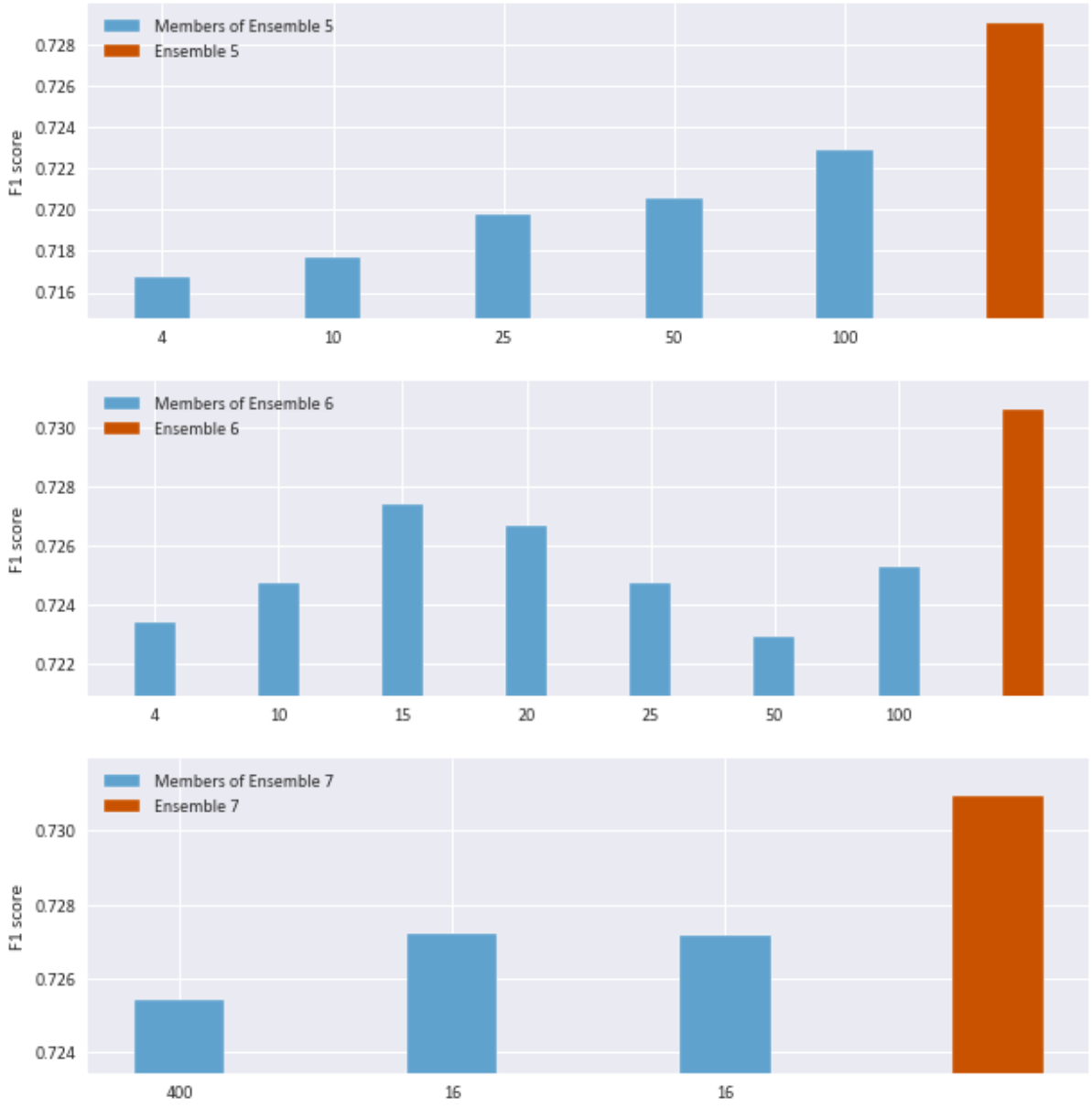


Εικόνα 5.18: Soft Vote Ensemble για Ensemble 3 και 4

Στη περίπτωση του stacking βρέθηκαν αρκετές ομάδες που να δίνουν καλύτερη απόδοση. Ενδεικτικά, χρησιμοποιήθηκαν οι εξής ομάδες:

	Γράφος	Μέγεθος παραθύρου	Μέθοδος
Ensemble 5	Μη Κατευθυνόμενος, χωρίς βάρη	4, 10, 25, 50, 100	-
Ensemble 6	Μη Κατευθυνόμενος, χωρίς βάρη	4, 10, 15, 20, 25, 50, 100	Rebase
Ensemble 7	Κατευθυνόμενος, χωρίς βάρη	400, 16, 16	Rebase και Μεταβλητό παράθυρο

Πίνακας 5.4: Ensembles για stacking στο 20NG



Εικόνα 5.19: Stacking Ensemble για Ensemble 5, 6 και 7

Παρατηρούμε τα εξής:

- Τα stacking ensembles είναι πιο ανεκτικά σε σχέση με τα χαρακτηριστικά που πρέπει να έχουν τα απλά μοντέλα της ομάδας. Επειδή λοιπόν το μοντέλο μαθαίνει ποια μοντέλα να εμπιστεύεται και πότε, μπορούμε να είμαστε πιο χαλαροί στην επιλογή των απλών μοντέλων. Φυσικά και πάλι είναι σημαντικό να έχουμε ποικιλία μοντέλων, ώστε τα λάθη να μην είναι συσχετισμένα, αλλά το περιθώριο λάθους είναι μεγαλύτερο στο stacking σε σχέση με το voting.
- Η βελτίωση στην απόδοση είναι σημαντική και κυμαίνεται από 0.03 έως 0.06 σε σύγκριση με το καλύτερο μοντέλο της ομάδας και σε μερικές περιπτώσεις ξεπερνάει το 0.73.
- Απαιτείται επιπλέον χρόνος για την κατασκευή του μετα-μοντέλου, αφού τα απλά μοντέλα πρέπει να προπονηθούν αρχικά σε όλα τα k-folds και έπειτα σε όλο το train set.

5.3 Αποτελέσματα στο Reuters8

Τα αποτελέσματα στο R8 θα παρουσιαστούν πιο συνοπτικά, επειδή είναι αρκετά παρόμοια με τα αντίστοιχα της προηγούμενης ενότητας. Οι κύριες διαφορές στο R8 σε σχέση με το 20NG είναι:

- τα κείμενα δεν είναι ισοκατανεμημένα στις κλάσεις, οπότε και χρησιμοποιήθηκε το $micro\ F_1\ score$ για την αξιολόγηση της ταξινόμησης
- η απόδοση είναι αρκετά μεγάλη και κυμαίνεται από 0.97 έως 0.98. Συνεπώς, το περιθώριο για βελτίωση είναι στενό.

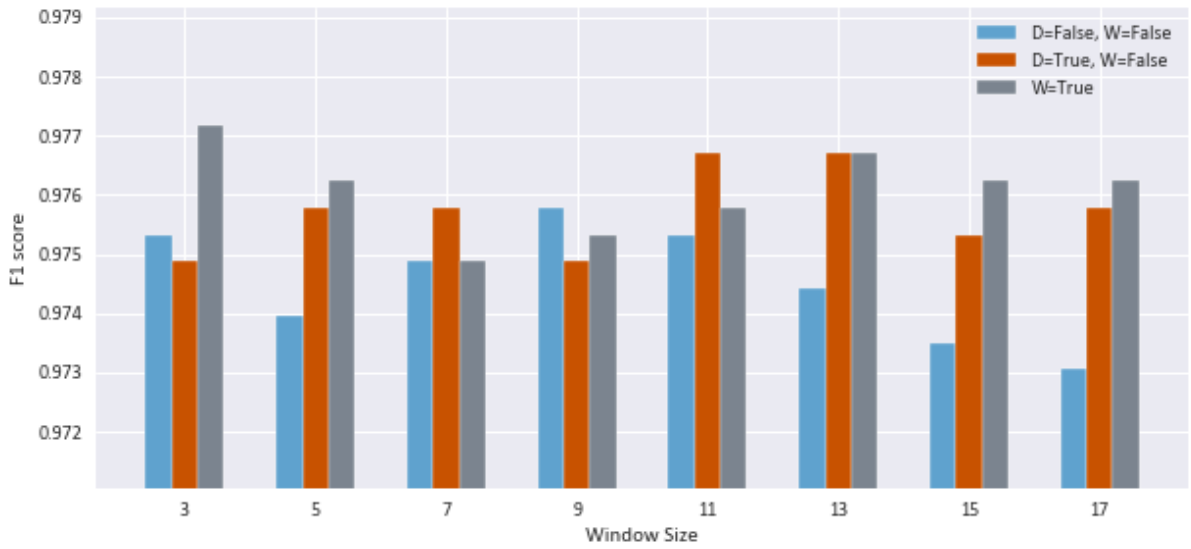
Για τους παραπάνω λόγους παρατηρούνται μερικές διαφορές στην απόδοση των μεθόδων, που θα αναφερθούν παρακάτω.

Βασικό Μοντέλο Γράφων Λέξεων

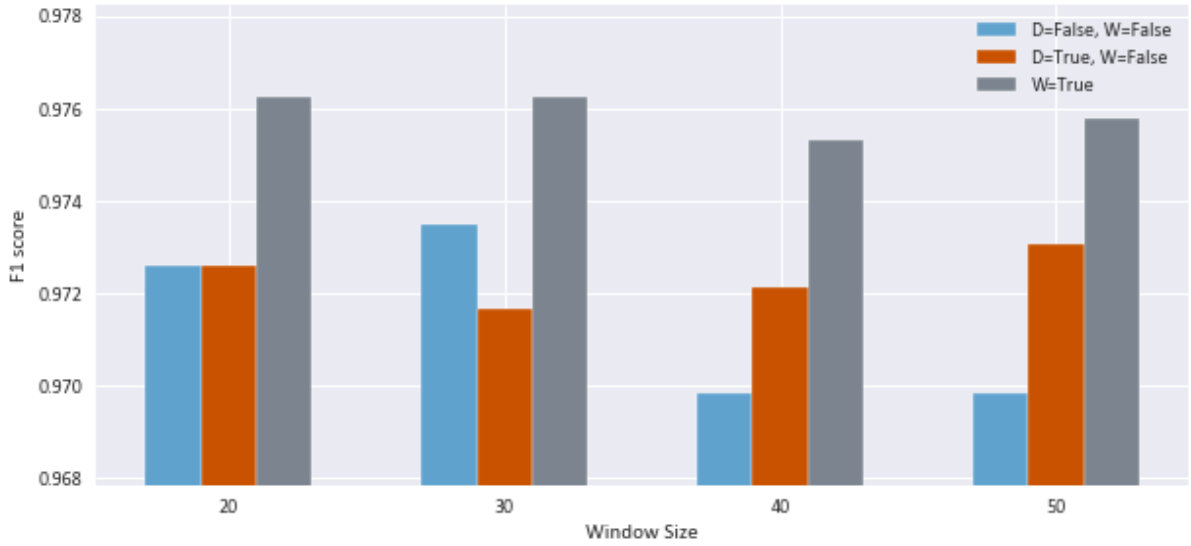
Τα αποτελέσματα για τους απλούς γράφους λέξεων απεικονίζονται στις Εικόνες 5.20 και 5.21. Παρατηρούμε ότι:

- Οι γράφοι λέξεων υπερτερούν και πάλι του TF-IDF, το οποίο έχει F_1 score ίσο με 0.9744 σε σύγκριση με τους γράφους λέξεων που πετυχαίνουν έως και 0.977.
- Η απόδοση των γράφων λέξεων είναι πολύ υψηλή ακόμα και για μικρά για μικρά παράθυρα και μάλιστα παίρνει μέγιστη τιμή 0.977 για μήκος παραθύρου 3 και γράφο με βάρη. Όταν μεγαλώνουμε το παράθυρο, η απόδοση εμφανίζει μικρή πτώση, εκτός από τους γράφους με βάρη οι οποίοι παραμένουν περίπου σταθεροί. Αυτό σημαίνει ότι το φαινόμενο της άνισης αντιπροσώπευσης δεν είναι τόσο έντονο σε αυτή τη συλλογή, κάτι που μπορεί να οφείλεται στο γεγονός ότι η πρώτη πρόταση στα κείμενα είναι ο τίτλος του άρθρου, άρα οι όροι αυτοί είναι πιθανό να υπάρχουν και πιο μετά στο κείμενο και άρα μειώνεται το φαινόμενο. Επιπλέον, τα κείμενα είναι πιο σύντομα και δεν χρειάζεται να χρησιμοποιηθεί πολύ μεγάλο μέγεθος παραθύρου.

- Οι γράφοι με βάρη είναι χρήσιμοι σε αυτή τη συλλογή κειμένων και όσο μεγαλώνει το μήκος παράθυρου ξεπερνάνε καθαρά σε απόδοση τους υπόλοιπους γράφους.



Εικόνα 5.20: Βασικό μοντέλο για μήκη παραθύρου 3-17



Εικόνα 5.21: Βασικό μοντέλο για μήκη παραθύρου 20-50

Coreference Resolution

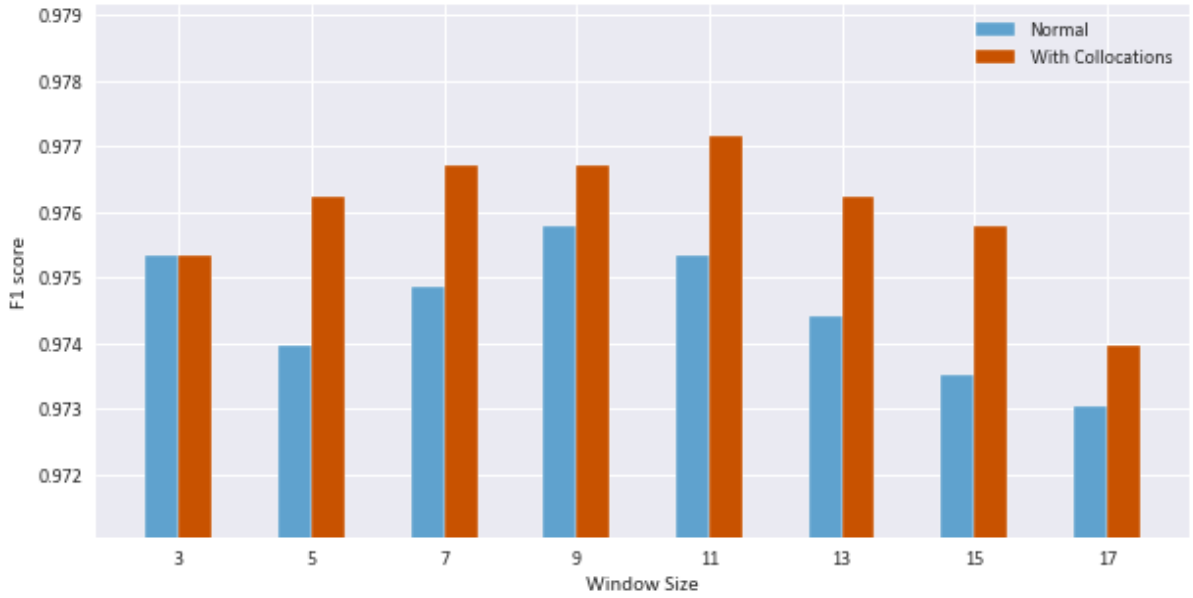
Τα αποτελέσματα για το coreference resolution δεν παρουσιάζονται, γιατί η έκδοση R8 που χρησιμοποιήθηκε είχε υποστεί προεπεξεργασία. Πιο συγκεκριμένα, έχουν αφαιρεθεί όλα τα σύμβολα στίξης και οι αριθμοί και έχουν χρησιμοποιηθεί μόνο μικρά γράμματα. Ως αποτέλεσμα, η είσοδος δεν είναι στη μορφη που αναμένει το μοντέλο coreference resolution και αποτυγχάνει να βρει χρήσιμες αναφορές.

Collocation Detection

Οι παράμετροι στην περίπτωση του R8 ορίστηκαν ως εξής:

- $\min count = 1$
- $threshold = 160$

Επιπλέον, η διαδικασία αναγνώρισης πραγματοποιήθηκε μόνο μια φορά και τα αποτελέσματα αντιστοιχούν σε μη κατευθυνόμενους γράφους χωρίς βάρη. Όπως φαίνεται στην Εικόνα 5.22, το collocation detection εξακολουθεί να βοηθά στην βελτίωση της απόδοσης και πετυχαίνει αύξηση από 0.1 έως 0.2 στο F_1 score.



Εικόνα 5.22: Collocation detection για min count = 1 και threshold = 160

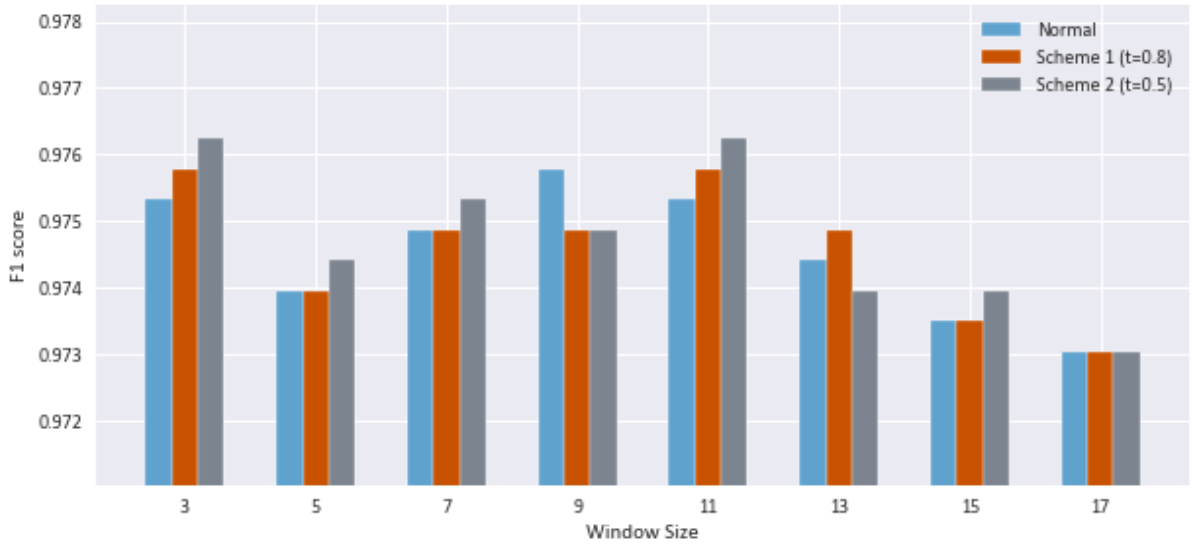
Word Embeddings

Λαμβάνοντας υπόψη τα αντίστοιχα αποτελέσματα της προηγούμενης ενότητας, τα σχήματα βάρους για την ανάθεση βάρους που εξετάστηκαν είναι τα εξής:

$$w_{i,j} = 1 + t * abs\left(sim\left(emb_i, emb_j\right)\right), \qquad για\ t = 0.8$$
$$w_{i,j} = t + (1 - t) * abs\left(sim\left(emb_i, emb_j\right)\right), \qquad για\ t = 0.5$$

Σε αντίθεση με πρίν, παρατηρούμε μια μικρή αύξηση της απόδοσης για ορισμένα μήκη παραθύρου και για τα δύο σχήματα ανάθεσης. Ωστόσο, η διαφορά αυτή είναι μικρότερη του 0.1 και δεν είναι σταθερή καθώς παρατηρούνται αρκετές αυξομειώσεις όταν αλλάζει το μέγεθος παραθύρου. Συνεπώς, η βελτίωση που παρατηρείται μπορεί εύκολα να αποδοθεί σε μικρές αλλαγές στη βαρύτητα των όρων που άλλωτε βοηθούν και άλλωτε όχι, στην καλύτερη ταξινόμηση κάποιων συγκεκριμένων κειμένων της συλλογής. Με άλλα λόγια, τα αποτελέσματα αυτής της ενότητας δεν είναι αρκετά ικανοποιητικά, ώστε να υπάρχει

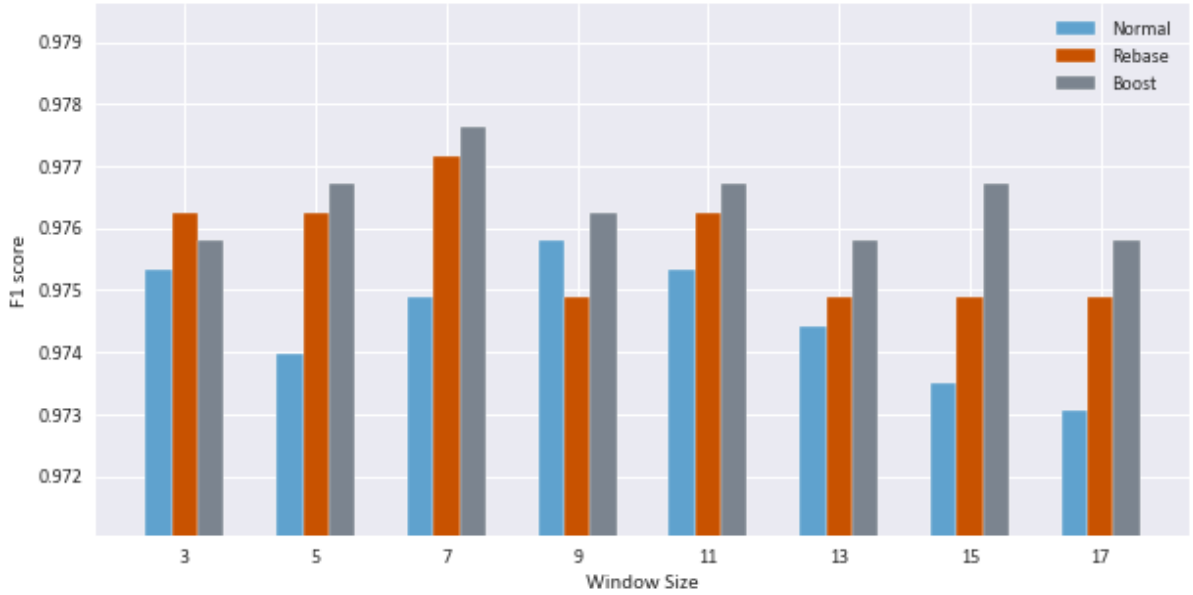
βεβαιότητα ότι η μέθοδος αυτή μπορεί να βελτιώσει την αναπαράσταση των όρων στους γράφους λέξεων.



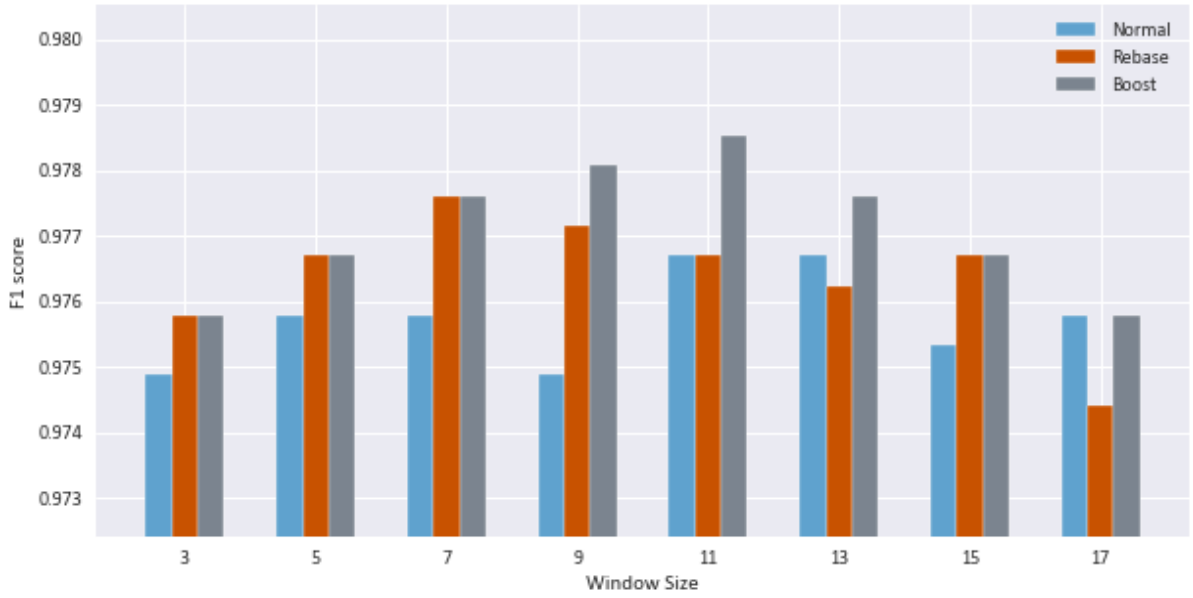
Εικόνα 5.23: Σχήμα ανάθεσης βάρους $w_{i,j} = 1 + t * abs\left(sim(emb_i, emb_j)\right)$ για $t = 0.8$ και $w_{i,j} = t + (1 - t) * abs\left(sim(emb_i, emb_j)\right)$ για $t = 0.5$

Ενίσχυση Κόμβων

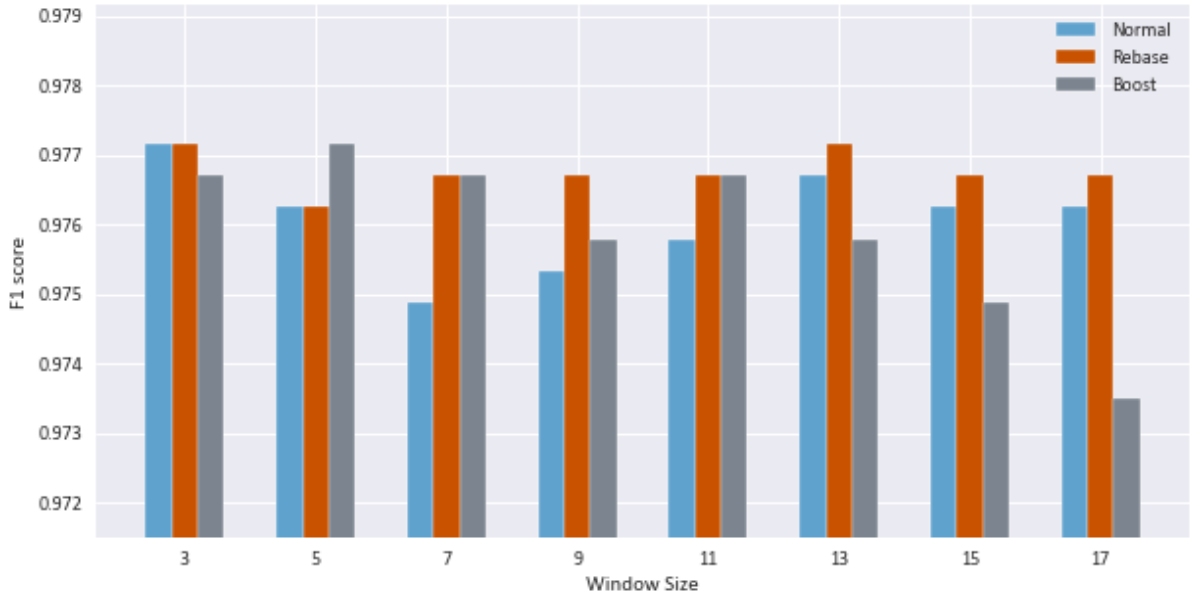
Τα αποτελέσματα για όλες τις διαμόρφωσεις των κόμβων φαίνονται στις Εικόνες 5.24-5.26. Παρατηρούμε ότι οι μέθοδοι Rebase και Boost είναι πιο ασταθές σε σχέση με πρίν και παρόλο που δεν παύει να υπάρχει αύξηση στην απόδοση, δεν είναι τόσο μεγάλη όσο στο 20NG. Το γεγονός αυτό συμφωνεί και με την παρατήρηση που έγινε στους απλούς γράφους, όπου δεν υπάρχει βελτίωση της απόδοσης στα πολύ μεγάλα παράθυρα. Συνεπώς, σε αυτήν τη συλλογή κειμένων το φαινόμενο άνισης αντιπροσώπευσης των ακριανών όρων δεν είναι τόσο έντονο. Σε κάθε περίπτωση, η μέθοδος ενίσχυσης κόμβων βοηθάει σε αρκετές περιπτώσεις στη βελτίωση και η απόδοση ξεπερνάει το 0.98 στους κατευθυνόμενους γράφους χωρίς βάρη. Η αύξηση στο F_1 score, όταν αυτή υπάρχει, μπορεί να φτάσει και το 0.3 αλλά συνήθως είναι γύρω στο 0.1.



Εικόνα 5.24: Rebase και Boost για μη κατευθυνόμενους γράφους χωρίς βάρος



Εικόνα 5.25: Rebase και Boost για κατευθυνόμενους γράφους χωρίς βάρος



Εικόνα 5.26: Rebase και Boost για γράφους με βάρος

Μεταβλητό Μέγεθος Παραθύρου

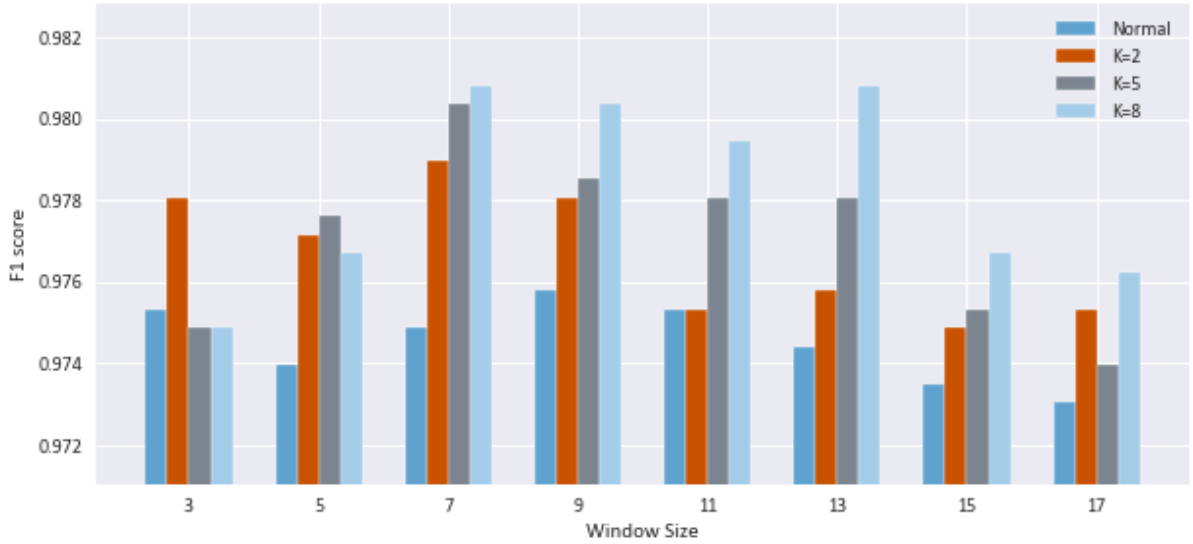
Για τη μέθοδο μεταβλητού μεγέθους παραθύρου εξετάστηκαν δύο τρόποι ανάθεσης των μεγεθών:

- Αύξηση στους πρώτους ws (window size) όρους, από ws σε $K * ws$, για $K = 2, 5, 8$
- Γραμμική μείωση από $8 * ws$ για τον πρώτο όρο μέχρι ws για τον τελευταίο και αντίστροφα γραμμική αύξηση από ws έως $8 * ws$

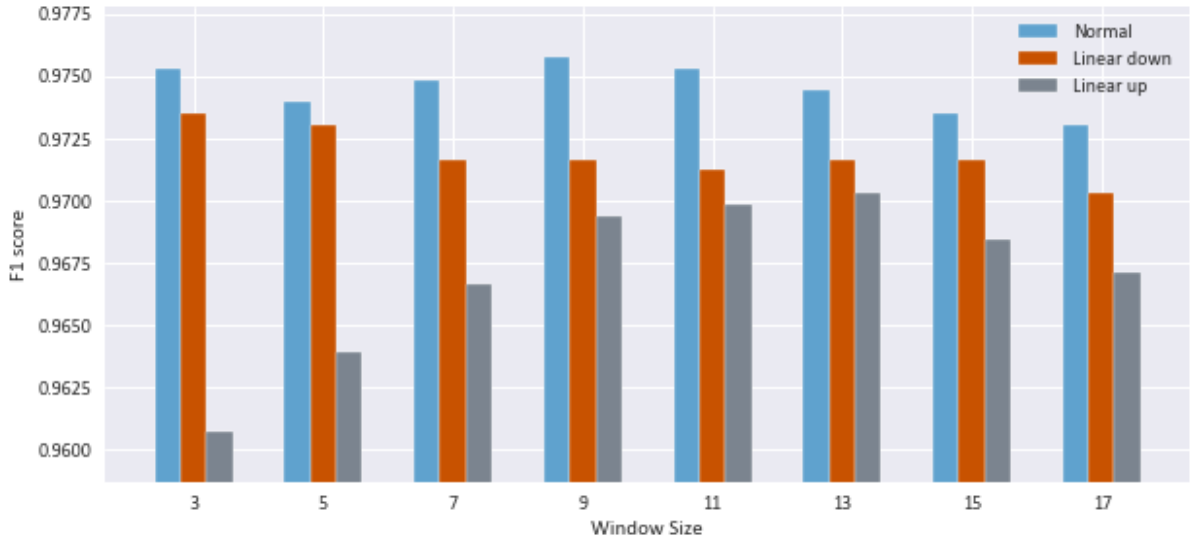
Τα αποτελέσματα παρουσιάζονται στις Εικόνες 5.27 και 5.28.

Οι παρατηρήσεις που είχαν γίνει στην αντίστοιχη ενότητα για το 20NG ισχύουν και εδώ. Η μέθοδος αυτή και πάλι δίνει τα καλύτερα αποτελέσματα, αφού καταφέρνει να πετύχει απόδοση ίση με 0.981. Επιπλέον, παρατηρούμε ότι για μικρά μήκη παραθύρου δεν βελτιώνει

την απόδοση αλλά καθώς μεγαλώνει το παράθυρο πετυχαίνει αύξηση από 0.2 έως 0.6 στο F_1 score. Όσο αφορά τη γραμμική αυξομείωση, ενώ πάλι δεν είναι χρήσιμή, μπορούμε να συμπεράνουμε ότι οι πρώτοι όροι είναι πιο σημαντικοί και σε αυτή τη συλλογή κειμένων.



Εικόνα 5.27: Μεταβολή μεγέθους παραθύρου στους πρώτους όρους με πολλαπλασιασμό



Εικόνα 5.28: Γραμμική μεταβολή μήκους παραθύρου από 8*ws για το πρώτο όρο μέχρι ws για τον τελευταίο και αντίστροφα

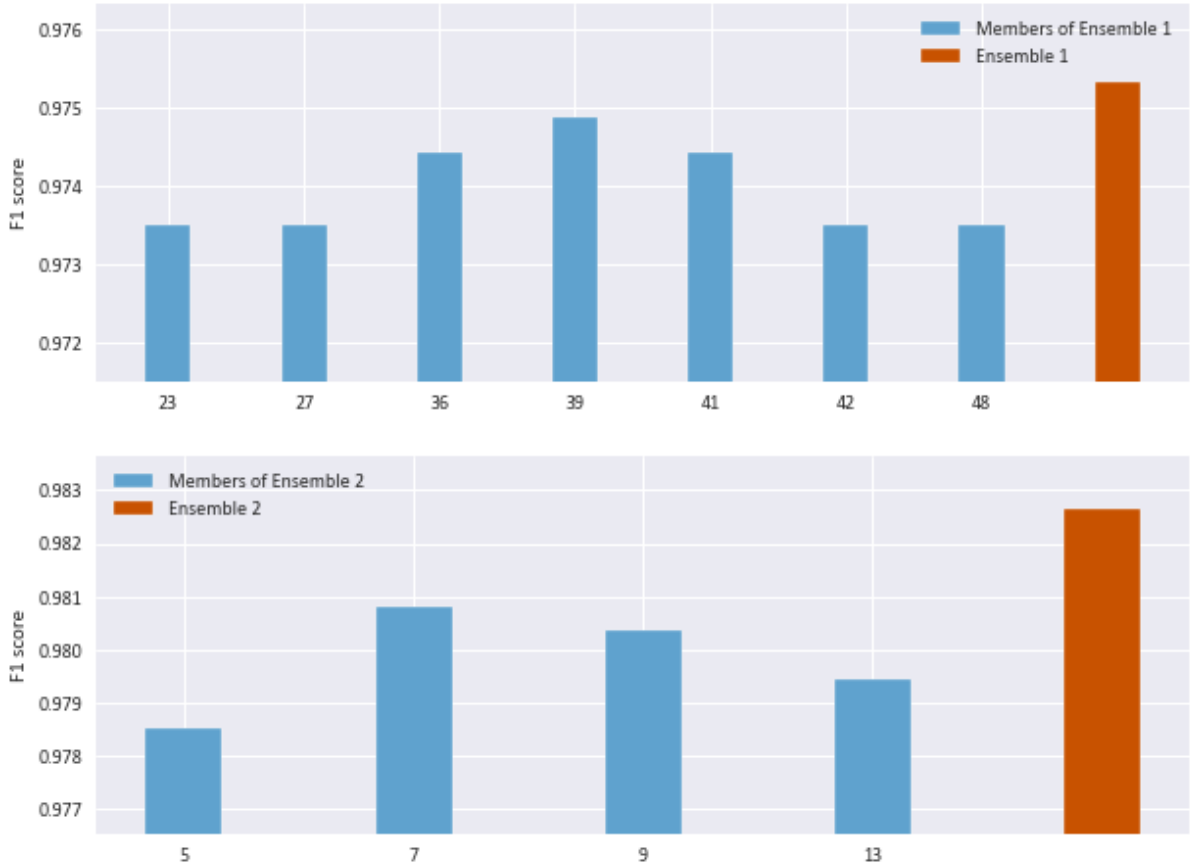
Ensembles Γράφων Λέξεων

Λόγω των αποτελεσμάτων στο 20NG, εξετάστηκε η χρήση μόνο των stacking ensembles, τα χαρακτηριστικά των οποίων φαίνονται στον πίνακα 5.5. Όπως έχουμε επισημάνει, για να είναι χρήσιμή η μέθοδος Ensembles θα πρέπει να μην υπάρχει συσχέτιση μεταξύ των λάθων των απλών μοντέλων. Στην περίπτωση του R8, αυτή η συνθήκη είναι δύσκολο να επιτευχθεί γιατί όλα τα μοντέλα πετυχαίνουν πολύ καλή απόδοση και τα κείμενα που τους δυσκολεύουν είναι παρόμοια. Ως επακόλουθο, η εύρεση ensembles ήταν πιο δύσκολή αυτή τη φορά σε σχέση με το 20NG, ακόμα και στη περίπτωση του stacking.

	Γράφος	Μέγεθος παραθύρου	Μέθοδος
Ensemble 1	Μη Κατευθυνόμενος, με βάρη	23, 27, 36, 39, 41, 42, 48	-
Ensemble 2	Μη Κατευθυνόμενος, με βάρη	5, 7, 9, 13	Μεταβλητό παράθυρο

Πίνακας 5.5: Ensembles για stacking στο R8

Τα αποτελέσματα παρουσιάζονται στην παρακάτω εικόνα, όπου παρατηρούμε και πάλι να υπάρχει βελτίωση στην απόδοση, η οποία όμως φυσιολογικά δεν είναι τόσο μεγάλη όσο στο 20NG.



Εικόνα 5.29: Stacking Ensemble για Ensemble 1, 2

Κεφάλαιο 6

Συμπεράσματα και Μελλοντικές Κατευθύνσεις

Σε αυτήν την εργασία μελετήθηκε το μοντέλο γράφων λέξεων και προτάθηκαν διάφορες τροποποιήσεις για την βελτίωση του. Από τα αποτελέσματα της πειραματικής διαδικασίας προκύπτει ότι όλες οι τροποποιήσεις μπορούν να προσφέρουν, σε διαφορετικό βαθμό η καθεμιά, στην καλύτερη αναπαράσταση του κειμένου από τους γράφους λέξεων. Σε σχέση με τις προηγούμενες εργασίες που χρησιμοποιούν τους γράφους λέξεων, προτιμήθηκε το all-degree για την ανάθεση βάρους στους όρους και δόθηκε μεγάλη προσοχή στο μήκος του παραθύρου ως παράμετρος, καθώς επηρεάζει αρκετά την απόδοση του μοντέλου. Συγκεκριμένα, ενώ αρχικά η απόδοση μειώνεται καθώς μεγαλώνει το παράθυρο, από ένα σημείο και μετά παρατηρείται σταδιακή αύξηση και το καλύτερο F_1 score προκύπτει για πολύ μεγάλα παράθυρα. Επιπλέον, όσο αφορά τον ίδιο το γράφο δεν παρατηρήθηκε κάποια προτιμητέα διαμόρφωση για τη κατεύθυνση των ακμών ή για τη χρήση βάρους σε αυτές.

Οι μέθοδοι προεπεξεργασίας coreference resolution και collocation detection μπορούν να βοηθήσουν τους γράφους λέξεων, αλλά χρειάζεται προσοχή στην επιλογή των παραμέτρων τους, αφού είναι αρκετά ευαίσθητες σε αλλαγές. Στην περίπτωση του coreference resolution επωφελούμαστε από αντικαταστάσεις αναφορών που καταλήγουν σε πολλούς όρους, καθώς δημιουργούνται αρκετές νέες χρήσιμες συνδέσεις.

Τα αποτελέσματα για τα word embeddings είναι διαφορετικά στα δύο σύνολα δεδομένων που χρησιμοποιήθηκαν. Στο 20NG, η χρήση της απόστασης ομοιότητας έδωσε αρκετά χειρότερα αποτελέσματα σε σχέση με τους απλούς γράφους λέξεων, ενώ αντίθετα στο R8 παρατηρείται μια μικρή βελτίωση σε ορισμένες περιπτώσεις. Σε κάθε περίπτωση, επειδή η επίδοση στο R8 είναι αρκετά μεγάλη, η αύξηση που παρατηρείται σε αυτό δεν είναι αρκετή ώστε να πείσει για τη χρησιμότητα της μεθόδου, η οποία μάλλον κρίνεται ανεπιτυχής.

Το πρόβλημα τις άνισης αντιπροσώπευσης των όρων που βρίσκονται στις άκρες του κειμένου είναι υπαρκτό, όπως προκύπτει από τα θετικά αποτελέσματα των μεθόδων ενίσχυσης των κόμβων, Rebase και Boost. Και οι δύο μέθοδοι βελτιώνουν σημαντικά την απόδοση των γράφων λέξεων, η οποία φτάνει ή ξεπερνάει το επίπεδο απόδοσης των γράφων με πολύ μεγάλα παράθυρα. Το αρνητικό με τη μέθοδο Rebase είναι ότι πρέπει να οριστεί το όριο κάτω από το οποίο δεν επιτρέπεται να είναι τα βάρη των όρων, οπότε στην περίπτωση όπου αυτό είναι δύσκολο να οριστεί, όπως στους γράφους με βάρους, η επίδραση της μεθόδους περιορίζεται.

Τα καλύτερα αποτελέσματα προέκυψαν με χρήση της μεθόδου μεταβλητού παραθύρου. Όπως φαίνεται από τα αποτελέσματα, υπάρχουν όροι μέσα στο κείμενο που επωφελούνται από ένα μεγαλύτερο παράθυρο και μάλιστα με τη μέθοδο αυτή μπορούμε σε κάποιο βαθμό να προσδιορίσουμε ποιοί είναι αυτοί οι όροι. Συγκεκριμένα, στα σύνολα δεδομένων που χρησιμοποιήθηκαν οι όροι που βρίσκονται στην αρχή των κειμένων είναι αρκετά σημαντικοί για το πρόβλημα της ταξινόμησης. Επιπλέον, η μέθοδος αυτή μπορεί να συνδυαστεί με τις μεθόδους ενίσχυσης κόμβων για περαιτέρω βελτίωση.

Τέλος, οι γράφοι λέξεων μπορούν εύκολα να χρησιμοποιηθούν για τον σχηματισμό ensembles, καθώς οι διάφορες επιλογές που υπάρχουν για τη δημιουργία των γράφων προσφέρουν την απαιτούμενη ποικιλία που χρειάζεται για την δημιουργία των ομάδων. Η πιο χρήσιμη μέθοδος ensemble που εξετάστηκε είναι το stacking και ακολουθούν το hard vote και το soft vote, που χρειάζονται μεγαλύτερη προσοχή στην επιλογή των μελών του ensemble. Στην περίπτωση του R8, όπου τα λάθη των απλών μοντέλων είναι αρκετά παρόμοια, η μέθοδος είναι λίγοτερο χρήσιμη σε σχέση με το 20NG.

Η δουλειά της διπλωματικής θα μπορούσε να επεκταθεί και να βελτιωθεί με ποικίλους τρόπους. Καταρχήν, θα ήταν χρήσιμο να αξιολογηθούν οι προτεινόμενες τροποποιήσεις και σε άλλους τομείς της φυσικής επεξεργασίας γλώσσας, ώστε να καθοριστεί αν είναι χρήσιμες μόνο για την ταξινόμηση κειμένων ή βελτιώνουν γενικά τους γράφους λέξεων. Μια άλλη ενδιαφέρουσα προσέγγιση που αξίζει να μελετηθεί είναι η χρήση νευρωνικών δικτύων για γράφους (Graph Neural Networks ή GNN). Τα νευρωνικά δίκτυα έχουν εφαρμοστεί με μεγάλη επιτυχία σε τομείς όπως η αναγνώριση εικόνας ή φωνής, όπου τα δεδομένα ανήκουν σε κάποιο ευκλείδειο χώρο, ωστόσο στο παρελθόν δεν είχαν την ανάλογη επιτυχία σε δεδομένα με πιο σύνθετη δομή, όπως οι γράφοι. Τα τελευταία χρόνια έχει σημειωθεί σημαντική πρόοδος στο τομέα της μάθησης πάνω στους γράφους, με μοντέλα όπως το GNN (Graph Neural Network), το GCNN (Graph Convolutional Neural Network) και διάφορες παραλλαγές αυτών, όπως παρουσιάζονται στο [28]. Στην περίπτωση μας, μπορούμε να χρησιμοποιήσουμε αυτές τις μεθόδους μάθησης πάνω στους γράφους λέξεων ως μια διαφορετική μέθοδο εξαγωγής χαρακτηριστικών για τους όρους. Μέχρι στιγμής, το βάρος κάθε όρου προέκυπτε από το βαθμό του αντίστοιχου κόμβου. ΄Ομως υπάρχει η δυνατότητα τα βάρη να προκύψουν μέσω μιας διαδικασίας μάθησης είτε μέσω των GNN είτε άλλων τεχνικών για την εξαγωγή node embeddings, ώστε να εκμεταλλευτούμε την ενδιάμεση αναπαράσταση των κειμένων ως γράφους.

Σχετικά με τις τροποποιήσεις, η πιο σημαντική επέκταση που πρέπει να διερευνηθεί είναι μια πιο συστηματική και αυτοματοποιημένη μέθοδος για τον καθορισμού του μεταβλητού παραθύρου. Η μέθοδος αυτή βελτίωσε σημαντικά την απόδοση των γράφων λέξεων, ωστόσο ο τρόπος που αναθέτουμε τα καινούργια μήκη παραθύρου βασίζεται σε ευριστικές και σε δοκιμές. Από εκεί και πέρα, η μέθοδος collocation detection θα μπορούσε να αντικατασταθεί από μια πιο γενική μέθοδο εύρεσης εκφράσεων, όπως το Named Entity Recognition, ώστε να προκύψουν μεγαλύτερες εκφράσεις που να να αντιπροσωπεύουν πιο σωστά τις διάφορες οντότητες του κειμένου. Το coreference resolution φάνηκε να δίνει καλά αποτελέσματα, όμως η επίδοση της βιβλιοθήκης που χρησιμοποιήθηκε δεν ήταν ικανοποιητική, οπότε εναλλακτικά στο μέλλον θα μπορούσαμε είτε να επικεντρωθούμε σε πιο απλές αναφορές είτε να χρησιμοποιηθεί μια διαφορετική βιβλιοθήκη για περαιτέρω βελτίωση της απόδοσης. Τέλος, αν και στη παρούσα εργασία η χρήση των word embeddings δεν φάνηκε πολύ χρήσιμη, θα ήταν πολύ χρήσιμο να εξεταστούν και άλλοι τρόποι χρησιμοποίησης τους για την κατασκευή των γράφων λέξεων, αφού τα word embeddings έχουν αποδειχτεί πολύ χρήσιμα για την επίλυση πολλών προβλημάτων του τομέα της επεξεργασίας φυσικής γλώσσας.

Βιβλιογραφία

[1] F. Rousseau and M. Vazirgiannis, “Graph-of-word and TW-IDF: new approach to ad hoc IR,” in *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 2013.

[2] A. Singhal, J. Choi, D. Hindle, D. D. Lewis and F. Pereira, “At&t at trec-7,” *NIST SPECIAL PUBLICATION SP,* pp. 239-252, 1999.

[3] J. Beel, S. Langer, M. Genzmehr, B. Gipp, C. Breitinger and A. Nürnberger, “Research paper recommender system evaluation: a quantitative literature survey,” in *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation*, 2013.

[4] K. Sparck Jones, “A statistical interpretation of term specificity and its application in retrieval,” *Journal of documentation,* vol. 28, pp. 11-21, 1972.

[5] T. Mikolov, K. Chen, G. Corrado and J. Dean, “Efficient estimation of word representations in vector space,” *arXiv preprint arXiv:1301.3781,* 2013.

[6] J. Pennington, R. Socher and C. Manning, “Glove: Global vectors for word representation,” in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014.

[7] S. S. Sonawane and P. A. Kulkarni, “Graph based representation and analysis of text document: A survey of techniques,” *International Journal of Computer Applications,* vol. 96, 2014.

[8] R. Mihalcea and P. Tarau, “Textrank: Bringing order into text,” in *Proceedings of the 2004 conference on empirical methods in natural language processing*, 2004.

[9] F. Rousseau and M. Vazirgiannis, “Main core retention on graph-of-words for single-document keyword extraction,” in *European Conference on Information Retrieval*, 2015.

[10] A. Tixier, F. Malliaros and M. Vazirgiannis, “A graph degeneracy-based approach to keyword extraction,” in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016.

[11] A. Tixier, P. Meladianos and M. Vazirgiannis, “Combining graph degeneracy and submodularity for unsupervised extractive summarization,” in *Proceedings of the workshop on new frontiers in summarization*, 2017.

[12] F. D. Malliaros and K. Skianis, “Graph-based term weighting for text categorization,” in *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, 2015.

[13] G. Bekoulis and F. Rousseau, “Graph-based term weighting scheme for topic modeling,” in *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, 2016.

[14] F. Rousseau, E. Kiagias and M. Vazirgiannis, “Text categorization as a graph classification problem,” in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2015.

[15] G. Nikolentzos, P. Meladianos, F. Rousseau, Y. Stavrakas and M. Vazirgiannis, “Shortest-path graph kernels for document similarity,” in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017.

[16] A. Tixier, K. Skianis and M. Vazirgiannis, “Gowvis: a web application for graph-of-words-based text visualization and summarization,” in *Proceedings of ACL-2016 System Demonstrations*, 2016.

[17] H. Levesque, E. Davis and L. Morgenstern, “The winograd schema challenge,” in *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*, 2012.

[18] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research,* vol. 12, pp. 2825-2830, 2011.

[19] M. Honnibal and I. Montani, “spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing,” *To appear,* 2017.

[20] R. Řehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, Valletta, 2010.

[21] A. Hagberg, P. Swart and D. S Chult, “Exploring network structure, dynamics, and function using NetworkX,” 2008.

[22] R. Weischedel, M. Palmer, M. Marcus, E. Hovy, S. Pradhan, L. Ramshaw, N. Xue, A. Taylor, J. Kaufman, M. Franchini and others, “Ontonotes release 5.0 ldc2013t19,” *Linguistic Data Consortium, Philadelphia, PA,* vol. 23, 2013.

[23] K. Clark and C. D. Manning, “Deep reinforcement learning for mention-ranking coreference models,” *arXiv preprint arXiv:1609.08667,* 2016.

[24] K. Clark and C. D. Manning, “Improving coreference resolution by learning entity-level distributed representations,” *arXiv preprint arXiv:1606.01323,* 2016.

[25] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado and J. Dean, “Distributed representations of words and phrases and their compositionality,” in *Advances in neural information processing systems*, 2013.

[26] G. Bouma, “Normalized (pointwise) mutual information in collocation extraction,” *Proceedings of GSCL,* pp. 31-40, 2009.

[27] A. Cardoso-Cachopo, *Improving Methods for Single-label Text Categorization,* 2007.

[28] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang and P. S. Yu, “A comprehensive survey on graph neural networks,” *arXiv preprint arXiv:1901.00596,* 2019.

[29] J. R. Firth, “A synopsis of linguistic theory, 1930-1955,” *Studies in linguistic analysis,* 1957.