



Πολυτεχνική Σχολή  
Τμήμα Μηχανικών Η/Υ και Πληροφορικής

Διπλωματική Εργασία

Σύστημα υποστήριξης αποφάσεων για  
την αγορά προϊόντων βασισμένο σε  
ανάλυση συναισθήματος

Ορέστης Μαραζιώτης  
AM: 1064028

Επιβλέπων:  
Χρήστος Μακρής  
Αναπληρωτής Καθηγητής

Πάτρα, Ιούλιος 2023



## Ευχαριστίες

Αυτό το κεφάλαιο στη ζωή μου δεν θα ήταν δυνατό χωρίς όλη τη συναισθηματική και επιστημονική υποστήριξη πολλών ανθρώπων που συνόδευσαν τη διαδρομή μου αυτά τα τελευταία χρόνια.

Καταρχάς, θα ήθελα να εκφράσω τις ειλικρινείς μου ευχαριστίες στον επιβλέποντα μου, καθηγητή κ. Χρήστο Μακρή, που μου έδωσε την ευκαιρία να εκπονήσω τη διπλωματική μου εργασία υπό την επίβλεψή του. Είμαι πολύ ευγνώμων για την ενθάρρυνση, την καθοδήγησή του και για την ευκαιρία να συνεργαστώ με την συγκεκριμένη ερευνητική ομάδα.

Είμαι επίσης πολύ ευγνώμων στον διδακτορικό φοιτητή Υποψήφιο Διδάκτωρ Αγοράκη Μπομπότα, για όλη τη γνώση, τον ενθουσιασμό και την υπομονή που μου επέτρεψε να πετύχω τους στόχους μου και να θέσω ορόσημα σε όλη την έρευνα και ανάπτυξη του έργου μου.

Τέλος, θα ήθελα να ευχαριστήσω όλη την οικογένειά μου καθώς ήταν οι άνθρωποι που έπαιξαν τον πιο σημαντικό ρόλο όλα αυτά τα χρόνια, με στήριξαν, πίστευαν σε μένα, θα είμαι πάντα ευγνώμων. Χωρίς την ψυχολογική τους υποστήριξη, δεν θα μπορούσα να ακολουθήσω τα όνειρα και τις προσδοκίες μου. Σας ευχαριστώ για όλα.



# Πίνακας περιεχομένων

Πίνακας περιεχομένων .....	i
Πίνακας εικόνων.....	iii
Περίληψη .....	vii
<b>Abstract.....</b>	x
<b>1. Εισαγωγή.....</b>	<b>1</b>
1.1. Σημασία του προβλήματος .....	1
1.2. Συνεισφορά της Διπλωματικής Εργασίας.....	1
1.3. Διάρθρωση της Διπλωματικής Εργασίας.....	2
<b>2. Ανάλυση Συναισθημάτων (Sentiment Analysis) .....</b>	<b>5</b>
2.1. Γιατί είναι χρήσιμη η Αναγνώριση Συναισθημάτων.....	6
2.2. Πώς λειτουργεί η Αναγνώριση Συναισθημάτων.....	6
2.2.1. Αναγνώριση Συναισθημάτων χωρίς τη χρήση Μηχανικής Μάθησης .....	7
2.2.2. Αναγνώριση Συναισθημάτων με τη χρήση Μηχανικής Μάθησης.....	10
2.3. Αξιολόγηση μοντέλων Ανάλυσης Συναισθημάτων.....	12
<b>3. Μηχανική Μάθηση.....</b>	<b>15</b>
3.1. Ιστορική αναδρομή .....	15
3.2. Βασικοί αλγόριθμοι Μηχανικής Μάθησης .....	17
3.2.1. Εκμάθηση με δέντρο απόφασης .....	17
3.2.2. Εκμάθηση με κανόνες συσχέτισης.....	18
3.2.3. Εκμάθηση με τεχνητά νευρωνικά δίκτυα .....	18
3.2.4. BERT .....	19
3.2.5. Βαθιά μάθηση.....	19
3.2.6. Transformers.....	20
3.2.7. Εκμάθηση με μηχανές διανυσμάτων υποστήριξης .....	21
3.2.8. Ομαδοποίηση .....	21
3.2.9. Δίκτυα Bayes .....	22
3.3. Παραδείγματα εφαρμογών .....	24
3.4. Ηθική.....	25
3.5. Λογισμικά που χρησιμοποιούνται ευρέως για Μηχανική Μάθηση .....	27
3.5.1. Scikit-learn .....	27
3.5.2. PyTorch .....	28
3.5.3. TensorFlow.....	28
3.5.4. Weka .....	29
3.5.5. Google Colab .....	31
3.5.6. Apache Mahout.....	33
3.5.7. Accord.Net .....	34
3.5.8. Shogun .....	35
3.5.9. Keras.io .....	36
3.5.10. Σύγκριση λογισμικών Μηχανικής Μάθησης.....	38
<b>4. Topic Modelling .....</b>	<b>41</b>

4.1.	<i>LDA topic modelling</i> τεχνική .....	42
4.1.1.	Βήμα προς βήμα παράδειγμα υλοποίησης του LDA (Latent Dirichlet Allocation) αλγορίθμου	43
4.2.	<i>BERTopic topic modelling</i> τεχνική .....	50
4.2.1.	Βήμα προς βήμα παράδειγμα υλοποίησης του BERTopic αλγορίθμου .....	51
5.	<b>Τεχνική Περιγραφή Υλοποίησης .....</b>	<b>63</b>
5.1.	Αρχιτεκτονική συστήματος .....	63
5.2.	<i>Web Scraper</i> .....	64
5.3.	<i>Datasets - Data Preparation</i> .....	65
5.4.	<i>Topic Modelling</i> .....	65
5.4.1.	LDA .....	65
5.4.2.	BERTopic .....	67
5.5.	Ανάλυση Συναισθήματος με Μηχανική Μάθηση .....	71
5.5.1.	Word2Vec .....	71
5.5.2.	Διαμέριση δοκιμαστικών, ελεγκτικών δεδομένων .....	72
5.5.3.	Hyperparameter Tuning.....	72
5.5.4.	Μετρικές Ακρίβειας .....	72
5.5.5.	Logistic Regression .....	73
5.5.6.	Decision Tree Classifier .....	73
5.5.7.	Bagging Classifier .....	74
5.5.8.	Random Forest Classifier.....	74
5.5.9.	Support Vector Classifier.....	75
5.5.10.	BERT .....	75
6.	<b>Επίλογος.....</b>	<b>79</b>
6.1.	Συμπεράσματα.....	79
6.2.	Μελλοντικές επεκτάσεις.....	80
7.	<b>Βιβλιογραφία .....</b>	<b>81</b>

## Πίνακας εικόνων

Εικόνα 2-1: Παράδειγμα Ανάλυσης Συναισθήματος. (Πηγή: <a href="https://www.gosmar.eu/machinelearning/2020/08/23/recurrent-neural-networks-for-sentiment-analysis/">https://www.gosmar.eu/machinelearning/2020/08/23/recurrent-neural-networks-for-sentiment-analysis/</a> ).....	5
Εικόνα 2-2: Η ρόδα του Plutchik. (Πηγή: <a href="https://martecgroup.com/using-plutchiks-wheel-of-emotions-in-market-research/">https://martecgroup.com/using-plutchiks-wheel-of-emotions-in-market-research/</a> ) .....	8
Εικόνα 2-3: Δισδιάστατο συναισθηματικό μοντέλο. (Shu et al., 2018) .....	9
Εικόνα 2-4: Τρισδιάστατο συναισθηματικό μοντέλο. (Shu et al., 2018) .....	10
Εικόνα 2-5: Παράδειγμα αναπαράστασης Bag-of-words για δυαδικές τιμές .....	12
Εικόνα 2-6: Παράδειγμα αναπαράστασης Bag-of-words για αριθμό εμφανίσεων. ...	12
Εικόνα 2-7: Κατηγοριοποίηση εφαρμόσιμων τεχνικών ανά προσέγγιση. ....	13
Εικόνα 3-1: Ιστορική Αναδρομή στη Μηχανική Μάθηση. (Πηγή: <a href="https://tinkeringchild.com/ideas-for-exploring-machine-learning-in-the-primary-years/">https://tinkeringchild.com/ideas-for-exploring-machine-learning-in-the-primary-years/</a> ).....	16
Εικόνα 3-2: Ένα γενικευμένο μοντέλο Μηχανικής Μάθησης. (Πηγή: <a href="http://repfiles.kallipos.gr/html_books/93/04a-main.html">http://repfiles.kallipos.gr/html_books/93/04a-main.html</a> ).....	17
Εικόνα 3-3: Παράδειγμα εκμάθησης με δέντρο απόφασης.....	18
Εικόνα 3-4: Παράδειγμα νευρωνικού δικτύου. ....	19
Εικόνα 3-5: Παράδειγμα Μηχανική Μάθησης και Βαθιάς Μάθησης.....	20
Εικόνα 3-6: Αρχιτεκτονική Transformer.....	20
Εικόνα 3-7: Απεικόνιση της αρχής της μάθησης με μηχανές διανυσμάτων υποστήριξης (SVM): (α) Ο χώρος εισόδου αντιστοιχίζεται στον χώρο χαρακτηριστικών με τη βοήθεια μιας συνάρτησης πυρήνα. (β) Διαχωρισμός υπερεπίπεδου και περιθωρίου για ταξινόμηση κατολισθήσεων.....	21
Εικόνα 3-8: Ταξινόμηση VS Ομαδοποίηση. ....	21
Εικόνα 3-9: Μηχανική μάθηση με επίβλεψη και χωρίς επίβλεψη - Αλγόριθμοι. (Πηγή: <a href="https://www.researchgate.net/publication/343022075_Automatic_recognition_of_handwritten_Arabic_characters_a_comprehensive_review/figures?lo=1">https://www.researchgate.net/publication/343022075_Automatic_recognition_of_handwritten_Arabic_characters_a_comprehensive_review/figures?lo=1</a> ).....	23
Εικόνα 3-10: Παράδειγμα ζητημάτων ηθικής σε εφαρμογές μηχανικής μάθησης στο χώρο της υγείας. Συγκεκριμένα: μεροληψία δεδομένων, ζητήματα ιδιωτικότητας, λογοδοσία και διαφάνεια, αξιοπιστία και εμπιστοσύνη. ....	25
Εικόνα 3-11: Επεξήγηση στη Μηχανική Μάθηση.....	26
Εικόνα 3-12: Περιβάλλον Scikit-learn. ....	27
Εικόνα 3-13: Παραδείγματα Scikit-learn. (Πηγή: <a href="https://scikit-learn.org/stable/auto_examples/index.html">https://scikit-learn.org/stable/auto_examples/index.html</a> ).....	27
Εικόνα 3-14: Παραδείγματα PyTorch. (Πηγή: <a href="https://pytorch.org/vision/stable/auto_examples/index.html">https://pytorch.org/vision/stable/auto_examples/index.html</a> ) .....	28

Εικόνα 3-15: Παράδειγμα περιβάλλοντος TensorFlow. (Πηγή: <a href="https://dzone.com/articles/tensorflow-simplified-examples?fromrel=true">https://dzone.com/articles/tensorflow-simplified-examples?fromrel=true</a> ) .....	29
Εικόνα 3-16: Παράδειγμα περιβάλλοντος Weka. (Πηγή: <a href="http://open.frapete.org/2013/09/weka-mooc-has-commenced/">http://open.frapete.org/2013/09/weka-mooc-has-commenced/</a> ) .....	31
Εικόνα 3-17: Παράδειγμα 1 στο περιβάλλον Colab. [38] .....	32
Εικόνα 3-18: Παράδειγμα 2 στο περιβάλλον Colab. [38] .....	32
Εικόνα 3-19: Παράδειγμα 3 στο περιβάλλον Colab. [38] .....	33
Εικόνα 3-20: Machine learning made in a minute / Μηχανική μάθηση στο λεπτό / Εφαρμογές Accord.NET (Πηγή: <a href="http://accord-framework.net/">http://accord-framework.net/</a> ) .....	35
Εικόνα 3-21: Στιγμότυπο οθόνης της εργαλειοθήκης SHOGUN. ....	36
Εικόνα 3-22: Αποτελέσματα έρευνας. (Πηγή: <a href="https://keras.io/why_keras/">https://keras.io/why_keras/</a> ) .....	37
Εικόνα 4-1: Παράδειγμα εφαρμογής Topic Modelling. (Πηγή: <a href="https://medium.com/analytics-vidhya/how-to-perform-topic-modeling-using-mallet-abc43916560f">https://medium.com/analytics-vidhya/how-to-perform-topic-modeling-using-mallet-abc43916560f</a> ) .....	41
Εικόνα 4-2: Παράδειγμα εφαρμογής LDA (Latent Dirichlet Allocation) αλγορίθμου. (Πηγή: <a href="http://chdoig.github.io/pytexas2015-topic-modeling/#/3/4">http://chdoig.github.io/pytexas2015-topic-modeling/#/3/4</a> ) .....	43
Εικόνα 4-3: Δείγμα ακατέργαστων δεδομένων. (Πηγή: <a href="https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0">https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0</a> ) .....	44
Εικόνα 4-4: Παράδειγμα αφαίρεση σημείων στίξεως. ....	45
Εικόνα 4-5: Παράδειγμα αφαίρεσης λέξεων τερματισμού. ....	46
Εικόνα 4-6: Εκπαίδευση μοντέλου LDA. ....	48
Εικόνα 4-7: Οπτικοποίηση αποτελεσμάτων μοντέλου LDA. ....	49
Εικόνα 4-8: Ρύθμιση σημειωματάριου και εισαγωγή των δεδομένων. ....	52
Εικόνα 4-9: Εισαγωγή και εξερεύνηση της φύσης του συνόλου δεδομένων. ....	53
Εικόνα 4-10: Οπτικοποίηση δεδομένων. ....	53
Εικόνα 4-11: Προεπεξεργασία/προετοιμασία δεδομένων. ....	55
Εικόνα 4-12: Προεπεξεργασία/προετοιμασία δεδομένων. ....	56
Εικόνα 4-13: Συχνότητα εμφάνισης θεμάτων. (Πηγή: <a href="https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8">https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8</a> ) .....	57
Εικόνα 4-14: Οπτικοποίηση του μοντέλου. (Πηγή: <a href="https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8">https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8</a> ) .....	58
Εικόνα 4-15: Topic Bar Chart with Topic Word Scores. (Πηγή: <a href="https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8">https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8</a> ) .....	59

Εικόνα 4-16: Ενσωματώσεις κειμένου σε 2 διαστάσεις. (Πηγή: <a href="https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8">https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8</a> ) .....	59
Εικόνα 4-17: Συγχύτητα εμφάνισης θεμάτων μετά τη μείωση των outliers. (Πηγή: <a href="https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8">https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8</a> ) .....	62
Εικόνα 5-1: Αρχιτεκτονική συστήματος. ....	63
Εικόνα 5-2: Snippet κώδικα για τη δημιουργία του web scraper.....	64
Εικόνα 5-3: Απεικόνιση προ-επεξεργασίας δεδομένων. ....	66
Εικόνα 5-4: Οπτικοποίηση σημαντικότερων λέξεων στις κριτικές (δεδομένα προς επεξεργασία).....	66
Εικόνα 5-5: Οπτικοποίηση των 5 κυρίαρχων topics σύμφωνα με τον αλγόριθμο LDA. .....	67
Εικόνα 5-6: Snippet κώδικα για τη χρήση του αλγορίθμου μηχανικής μάθησης Word2vec. ....	71
Εικόνα 5-7: Μετρικές αλγορίθμου Logistic Regression. ....	73
Εικόνα 5-8: finetuning και επιλογή hyperparameters. ....	76
Εικόνα 5-9: Σύγκλιση epochs.....	77



## Περίληψη

Η παρούσα διπλωματική εργασία επιδιώκει να επιλύσει ένα πρόβλημα που αντιμετωπίζουν οι καταναλωτές στη σύγχρονη αγορά προϊόντων. Η πληθώρα των επιλογών που παρέχονται στους καταναλωτές δημιουργεί πολυπλοκότητα και ανταγωνιστικότητα, καθώς οι αποφάσεις αγοράς επηρεάζονται από παραμέτρους όπως η ποιότητα, η τιμή, οι προσωπικές προτιμήσεις και οι συναισθηματικές αντιδράσεις. Η δυνατότητα των καταναλωτών να λαμβάνουν αποφάσεις αγοράς που ανταποκρίνονται στις ατομικές τους ανάγκες και προτιμήσεις είναι ζωτικής σημασίας.

Ο στόχος της εργασίας είναι η ανάπτυξη ενός συστήματος υποστήριξης αποφάσεων βασισμένου στην ανάλυση συναισθημάτων, το οποίο παρέχει την απαραίτητη πληροφορία στους καταναλωτές για να λαμβάνουν ενημερωμένες αποφάσεις αγοράς. Το σύστημα αυτό επιτρέπει στους καταναλωτές να αντιμετωπίζουν τον όγκο των διαθέσιμων προϊόντων και να επιλέγουν αυτά που ικανοποιούν καλύτερα τις ανάγκες και τις προτιμήσεις τους, προσφέροντας μεγαλύτερη ικανοποίηση.

Για την υλοποίηση του συστήματος, χρησιμοποιήθηκαν αλγόριθμοι μηχανικής μάθησης, καθώς αποδείχθηκε ότι προσφέρουν καλύτερη απόδοση και ακρίβεια στην αναγνώριση συναισθημάτων. Η εργασία εξετάζει διάφορες προσεγγίσεις για την αναγνώριση συναισθημάτων, με τη χρήση κανόνων, προκαθορισμένων μοντέλων και αλγορίθμων μηχανικής μάθησης. Η χρήση αλγορίθμων μηχανικής μάθησης επιτυγχάνει καλύτερη ακρίβεια στην αναγνώριση συναισθημάτων. Η υλοποίηση του συστήματος βασίστηκε στις βιβλιοθήκες Scikit-learn, TensorFlow και Keras.io, που παρέχουν ισχυρά εργαλεία και πλαίσια για την ανάπτυξη αλγορίθμων μηχανικής μάθησης.

Η εργασία παρουσιάζει τη διαδικασία ανάπτυξης του συστήματος υποστήριξης αποφάσεων βασισμένου στην ανάλυση συναισθημάτων. Κάθε κεφάλαιο παρουσιάζει μια συνολική άποψη για το πρόβλημα που επιδιώκεται να επιλυθεί, τις χρησιμοποιούμενες μεθόδους και τα επιτεύγματα. Αναλύονται οι διάφοροι αλγόριθμοι και τεχνικές που χρησιμοποιούνται για την αναγνώριση συναισθημάτων, ενώ παρουσιάζεται επίσης μια εισαγωγή στη μηχανική μάθηση και την εφαρμογή της στην ανάλυση συναισθημάτων. Επιπλέον, παρέχονται λεπτομερείς περιγραφές της αρχιτεκτονικής του συστήματος και των χρησιμοποιούμενων τεχνολογιών για τη διεκπεραίωση τη παρούσας διπλωματικής εργασίας.

Τέλος, παρουσιάζονται τα συμπεράσματα της εργασίας και προτείνονται μελλοντικές επεκτάσεις και κατευθύνσεις για περαιτέρω έρευνα στον τομέα της ανάλυσης συναισθημάτων με χρήση αλγορίθμων μηχανικής μάθησης. Η παρούσα διπλωματική εργασία επιτρέπει στον αναγνώστη να αποκτήσει μια πλήρη κατανόηση του προβλήματος που αντιμετωπίζεται και των προτεινόμενων λύσεων με τη χρήση αλγορίθμων μηχανικής μάθησης. Η διπλωματική εργασία διαρθρώνεται ως εξής:

- **Κεφάλαιο 1:** Εισαγωγή, όπου παρουσιάζεται το πρόβλημα, η σημασία του και η μεθοδολογία που χρησιμοποιήθηκε για την επίλυσή του. Επίσης, περιγράφεται η δομή της εργασίας.
- **Κεφάλαιο 2:** Ανάλυση των αλγορίθμων και τεχνικών αναγνώρισης συναισθημάτων, με εστίαση σε μηχανική μάθηση και τις χρησιμοποιούμενες βιβλιοθήκες.
- **Κεφάλαιο 3:** Εισαγωγή στη μηχανική μάθηση, περιγραφή των βασικών αλγορίθμων και μεθόδων, παραδείγματα εφαρμογών και συζήτηση ηθικών ζητημάτων.

- **Κεφάλαιο 4:** Περιγραφή των τεχνικών Topic Modelling που χρησιμοποιήθηκαν για την ανάλυση συναισθημάτων στην εργασία.
- **Κεφάλαιο 5:** Περιγραφή της αρχιτεκτονικής του συστήματος και των τεχνολογιών που χρησιμοποιήθηκαν, ανάλυση πειραμάτων και παρουσίαση αποτελεσμάτων αξιολόγησης.
- **Κεφάλαιο 6:** Συμπεράσματα, μελλοντικές επεκτάσεις και προτάσεις για περαιτέρω έρευνα στον τομέα της ανάλυσης συναισθημάτων με χρήση αλγορίθμων μηχανικής μάθησης.

Επομένως, η διπλωματική εργασία προσφέρει μια ολοκληρωμένη μελέτη σχετικά με την ανάπτυξη ενός συστήματος υποστήριξης αποφάσεων βασισμένου στην ανάλυση συναισθημάτων. Οι αναγνώστες μπορούν να αποκτήσουν μια ευρεία κατανόηση του προβλήματος, των προτεινόμενων λύσεων και των μεθόδων που χρησιμοποιούνται, καθώς και να εξετάσουν την αποτελεσματικότητα και την ακρίβεια του συστήματος.

**Λέξεις κλειδιά:** αγορά προϊόντων, ανάλυση συναισθήματος, μηχανική μάθηση, web scraper, topic modelling, LDA, BERT



## Abstract

This thesis seeks to resolve a problem faced by consumers in the modern product market. The plethora of choices available to consumers creates complexity and competitiveness, as purchasing decisions are influenced by parameters such as quality, price, personal preferences and emotional reactions. The ability of consumers to make purchasing decisions that meet their individual needs and preferences is vital.

The aim of the work is to develop a decision support system based on sentiment analysis, which provides the necessary information to consumers to make informed purchase decisions. This system allows consumers to face the volume of products available and choose those that best meet their needs and preferences, offering greater satisfaction.

To implement the system, machine learning algorithms were used, as they were proven to offer better performance and accuracy in sentiment analysis. This work examines various approaches to sentiment analysis, using rules, predefined models and machine learning algorithms. Nevertheless, it is proven that using machine learning algorithms achieves better accuracy in sentiment analysis. The system implementation was based on the Scikit-learn, TensorFlow and Keras.io libraries, which provide powerful tools and frameworks for developing machine learning algorithms.

This work presents the development process of the decision support system based on sentiment analysis. Each chapter presents an overall view of the problem sought to be solved, the methods used and the achievements. The various algorithms and techniques used for sentiment analysis are discussed, and an introduction to machine learning and its application to sentiment analysis is also presented. In addition, detailed descriptions of the system architecture and the technologies used to complete this thesis are provided.

Finally, the conclusions of this thesis are presented and future extensions and directions for further research in the field of sentiment analysis using machine learning algorithms are suggested. This thesis allows the reader to gain a thorough understanding of the problem at hand and the proposed solutions using machine learning algorithms. The thesis is structured as follows:

- **Chapter 1:** Introduction, where the problem is presented, its importance and the methodology used to solve it. Also, the structure of the work is described.
- **Chapter 2:** Analysis of sentiment analysis algorithms and techniques, with a focus on machine learning and the libraries used.
- **Chapter 3:** Introduction to machine learning, description of basic algorithms and methods, examples of applications, and discussion of ethical issues.
- **Chapter 4:** Description of Topic Modeling techniques used for sentiment analysis at work.
- **Chapter 5:** Description of the system architecture and technologies used, analysis of experiments and presentation of evaluation results.
- **Chapter 6:** Conclusions, future extensions and suggestions for further research in the field of sentiment analysis using machine learning algorithms.

Therefore, the thesis offers a comprehensive study on the development of a decision support system based on sentiment analysis. Readers can gain a broad understanding of the problem, the proposed solutions, and the methods used, as well as examine the efficiency and accuracy of the system.

**Keywords:** product buying, sentiment analysis, machine learning, web scraper, topic modeling, LDA, BERT



## 1. Εισαγωγή

Στο κεφάλαιο αυτό θα παρουσιάσουμε το πρόβλημα που η παρούσα διπλωματική εργασία σκοπεύει να επιλύσει, καθώς επίσης και τη σημασία του στην σημερινή εποχή. Έπειτα, θα εξηγήσουμε την προσέγγιση και μεθοδολογία μας προς την επίλυση του συγκεκριμένου προβλήματος. Τέλος, θα παρουσιάσουμε αναλυτικά τη διάρθρωση της συγκεκριμένης διπλωματικής εργασίας.

### 1.1. Σημασία του προβλήματος

Η σύγχρονη αγορά προϊόντων έχει εξελιχθεί σε έναν πολύπλοκο και ανταγωνιστικό χώρο, με αμέτρητες επιλογές για τους καταναλωτές. Κατά την αγορά προϊόντων, οι καταναλωτές αντιμετωπίζουν πολλές παραμέτρους που επηρεάζουν τις αποφάσεις τους, όπως η ποιότητα, η τιμή, οι προσωπικές προτιμήσεις και οι συναισθηματικές αντιδράσεις. Συνεπώς, η ικανότητα των καταναλωτών να λαμβάνουν αποφάσεις αγοράς που ανταποκρίνονται στις ατομικές τους ανάγκες και προτιμήσεις είναι ζωτικής σημασίας.

Το πρόβλημα της αγοράς προϊόντων αποτελεί ένα σημαντικό ζήτημα που αφορά τόσο τους καταναλωτές όσο και τις επιχειρήσεις. Η ανάπτυξη ενός συστήματος υποστήριξης αποφάσεων σε αυτό το πλαίσιο έχει σημαντικό αντίκτυπο και αξία για πολλούς λόγους:

- (1) Πληθώρα επιλογών: Στη σύγχρονη αγορά, οι καταναλωτές αντιμετωπίζουν μια τεράστια ποικιλία προϊόντων και υπηρεσιών που είναι διαθέσιμα. Αυτή η υπερβολική επιλογή μπορεί να καθιστά δύσκολη την εύρεση του καταλληλότερου προϊόντος. Ένα σύστημα υποστήριξης αποφάσεων μπορεί να παρέχει πληροφορίες και συστάσεις που θα διευκολύνουν την επιλογή του καταναλωτή.
- (2) Πολυπλοκότητα προϊόντων: Οι σύγχρονες αγορές προσφέρουν προϊόντα με υψηλή τεχνολογία και πολυπλοκότητα, όπως ηλεκτρονικές συσκευές, αυτοκίνητα, λογισμικό και πολλά άλλα. Η απόφαση αγοράς αυτών των προϊόντων είναι σημαντική και συνήθως απαιτεί εξειδικευμένες γνώσεις. Ένα σύστημα υποστήριξης αποφάσεων μπορεί να παρέχει τις απαραίτητες πληροφορίες και αναλύσεις για να βοηθήσει τον καταναλωτή να κατανοήσει και να επιλέξει το προϊόν που ταιριάζει καλύτερα στις ανάγκες του.
- (3) Επιρροή συναισθημάτων: Στις αγορές, τα συναισθήματα παίζουν σημαντικό ρόλο στην απόφαση αγοράς. Οι καταναλωτές συχνά επηρεάζονται από συναισθήματα όπως η ευχαρίστηση, η δυσαρέσκεια, ο φόβος ή η απόλαυση. Η ανάλυση συναισθημάτων μπορεί να προσφέρει πολύτιμες πληροφορίες για την αντίδραση των καταναλωτών σε ένα προϊόν ή μια υπηρεσία, και να συμβάλει στη βελτίωση της εμπειρίας αγοράς.

Οι παραπάνω λόγοι καθιστούν την ανάπτυξη ενός συστήματος υποστήριξης αποφάσεων βασισμένου στην ανάλυση συναισθημάτων σημαντική. Αυτό το σύστημα μπορεί να παρέχει αντικειμενικές και αξιόπιστες πληροφορίες στους καταναλωτές, επιτρέποντάς τους να λαμβάνουν ενημερωμένες αποφάσεις αγοράς. Επιπλέον, μπορεί να συμβάλει στη βελτίωση της εμπειρίας αγοράς και στην ενίσχυση της ανταγωνιστικότητας των επιχειρήσεων.

### 1.2. Συνεισφορά της Διπλωματικής Εργασίας

Στόχος αυτής της εργασίας είναι η ανάπτυξη ενός συστήματος υποστήριξης αποφάσεων που βασίζεται στην ανάλυση συναισθημάτων, με σκοπό να παρέχει την κατάλληλη πληροφορία στους καταναλωτές για τη λήψη ενημερωμένων αποφάσεων αγοράς. Αυτό το σύστημα θα τους δώσει τη δυνατότητα να αντιμετωπίζουν τον όγκο των προϊόντων που είναι διαθέσιμα και να

επιλέγουν αυτά που θα ικανοποιήσουν καλύτερα τις ανάγκες και τις προτιμήσεις τους και θα προσφέρουν τη μεγαλύτερη ικανοποίηση.

Για να επιτευχθεί αυτός ο στόχος, μελετήσαμε δύο διαφορετικές προσεγγίσεις για την αναγνώριση συναισθημάτων. Η πρώτη προσέγγιση αφορά τη χρήση αλγορίθμων αναγνώρισης συναισθημάτων χωρίς τη χρήση μηχανικής μάθησης. Αυτή η προσέγγιση εξετάζει τη χρήση κανόνων και προκαθορισμένων μοντέλων για την ανίχνευση συναισθημάτων σε κείμενα και άλλα αποτελέσματα. Ωστόσο, καταλήξαμε στο συμπέρασμα ότι η χρήση αλγορίθμων μηχανικής μάθησης προσφέρει καλύτερη απόδοση και ακρίβεια στην αναγνώριση συναισθημάτων. Για την υλοποίηση του συστήματος, χρησιμοποιήσαμε τις βιβλιοθήκες Scikit-learn, TensorFlow και Keras.io. Αυτές οι βιβλιοθήκες προσφέρουν ισχυρά εργαλεία και πλαίσια για την ανάπτυξη αλγορίθμων μηχανικής μάθησης και βαθιάς μάθησης. Χρησιμοποιώντας αυτές τις εργαλειοθήκες, μπορέσαμε να εκπαιδεύσουμε μοντέλα μηχανικής μάθησης που δύνανται να αναγνωρίσουν συναισθήματα από δεδομένα κειμένου, ήχου ή εικόνας. Η επιλογή αυτών των βιβλιοθηκών βασίστηκε στην αξιοπιστία, την ευελιξία και την ευκολία χρήσης που προσφέρουν.

Σε αυτήν την εργασία, ο αναγνώστης θα έχει την ευκαιρία να εξερευνήσει τη διαδικασία ανάπτυξης του συστήματος υποστήριξης αποφάσεων βασισμένου στην ανάλυση συναισθήματος. Σε κάθε κεφάλαιο, θα παρουσιάσουμε μια συνολική άποψη για το πρόβλημα που λύνουμε και τον σκοπό της εκάστοτε ανάλυσης. Θα εξηγήσουμε τις μεθόδους που χρησιμοποιήσαμε για την αναγνώριση συναισθημάτων και θα παρουσιάσουμε τα αποτελέσματα που προέκυψαν από την εφαρμογή των αλγορίθμων. Τέλος, θα αξιολογήσουμε τα αποτελέσματα και θα συζητήσουμε τις πιθανές επεκτάσεις και βελτιώσεις που μπορούν να γίνουν στο μέλλον. Με την ολοκλήρωση αυτής της εργασίας, αναμένεται ότι ο αναγνώστης θα έχει αποκτήσει μια πλήρη κατανόηση του προβλήματος που αντιμετωπίζουμε και των προτεινόμενων λύσεων που αναπτύξαμε με τη χρήση αλγορίθμων μηχανικής μάθησης.

### 1.3. Διάρθρωση της Διπλωματικής Εργασίας

Η διάρθρωση της παρούσας διπλωματικής εργασίας είναι η ακόλουθη:

- **Κεφάλαιο 1:** Σε αυτό το κεφάλαιο παρουσιάζουμε το πεδίο της συγκεκριμένης εργασίας, αναλύουμε το πρόβλημα που επιδιώκουμε να επιλύσουμε, ενώ παράλληλα εξηγούμε γιατί είναι σημαντικό. Περιγράφουμε επίσης τους στόχους και τις αναμενόμενες συνέπειες της έρευνάς μας. Τέλος, αναφέρουμε συνοπτικά τη μεθοδολογία που χρησιμοποιήσαμε για την ανάλυση του προβλήματος.
- **Κεφάλαιο 2:** Σε αυτό το κεφάλαιο αναλύουμε τους διάφορους αλγορίθμους και τεχνικές που χρησιμοποιούνται για την αναγνώριση συναισθημάτων. Ξεκινάμε περιγράφοντας μεθόδους αναγνώρισης συναισθημάτων που δεν χρησιμοποιούν μηχανική μάθηση, όπως κανόνες και λεξικογραφικές προσεγγίσεις. Στη συνέχεια, περιγράφουμε τη χρήση αλγορίθμων μηχανικής μάθησης για την αναγνώριση συναισθημάτων, εστιάζοντας σε εργαλεία και βιβλιοθήκες όπως το Scikit-learn, το TensorFlow και το Keras.io τα οποία χρησιμοποιήθηκαν για τη διεκπεραίωση της παρούσας διπλωματικής εργασίας.
- **Κεφάλαιο 3:** Σε αυτό το κεφάλαιο παρουσιάζουμε μια εισαγωγή στη μηχανική μάθηση. Εξηγούμε τι είναι η μηχανική μάθηση, ποιοι είναι οι βασικοί αλγόριθμοι και μέθοδοι που χρησιμοποιούνται, και παρουσιάζουμε παραδείγματα εφαρμογών της μηχανικής μάθησης στον τομέα της ανάλυσης συναισθημάτων αλλά και άλλους τομείς γενικότερα. Εξετάζουμε επίσης ζητήματα ηθικής που σχετίζονται με τη χρήση της μηχανικής μάθησης και παρουσιάζουμε τα λογισμικά που ευρέως χρησιμοποιούνται

για την υλοποίηση αλγορίθμων μηχανικής μάθησης, όπως το Scikit-learn, το TensorFlow και το Keras.io.

- *Κεφάλαιο 4:* Σε αυτό το κεφάλαιο περιγράφουμε τις τεχνικές Topic Modelling που χρησιμοποιήσαμε για τη διεκπεραίωση της παρούσας διπλωματικής εργασίας. Συγκεκριμένα, περιγράφουμε τον αλγόριθμο Lda και τη μέθοδο Bertopic. Εξηγούμε τον τρόπο λειτουργίας τους και πώς αξιοποιήθηκαν για την ανάλυση συναισθημάτων στην εργασία.
- *Κεφάλαιο 5:* Σε αυτό το κεφάλαιο περιγράφουμε αναλυτικά την αρχιτεκτονική του συστήματος που χρησιμοποιήσατε για την υλοποίηση της διπλωματικής εργασίας. Συγκεκριμένα, περιγράφουμε κάθε μέρος της αρχιτεκτονικής και τις τεχνολογίες που χρησιμοποιήσαμε. Επίσης, παρουσιάζουμε πειράματα, αποσπάσματα κώδικα, μετρικές και πίνακες που σχετίζονται με την υλοποίηση του συστήματος, καθώς επίσης και αποτελέσματα αξιολόγησης της απόδοσής του.
- *Κεφάλαιο 6:* Στο κεφάλαιο αυτό παρουσιάζουμε τα συμπεράσματα της εργασίας και τις μελλοντικές επεκτάσεις που μπορούν να γίνουν στο πεδίο της ανάλυσης συναισθημάτων με χρήση αλγορίθμων μηχανικής μάθησης. Αναφέρουμε επίσης πιθανές προκλήσεις και προτείνουμε μελλοντικές κατευθύνσεις για περαιτέρω έρευνα.



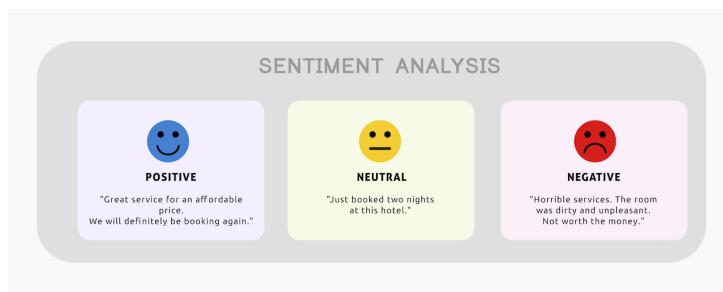
## 2. Ανάλυση Συναισθημάτων (Sentiment Analysis)

Η ανάλυση συναισθήματος, γνωστή επίσης ως αναγνώριση συναισθημάτων ή τεχνητή νοημοσύνη συναισθήματος, αποτελεί μια τεχνική επεξεργασίας φυσικής γλώσσας (Natural Language Processing - NLP) που αξιολογεί εάν ένα κομμάτι περιεχομένου εκφράζει θετική, αρνητική ή ουδέτερη στάση. Με την ανάλυση του κειμένου και τη χρήση στατιστικών δεδομένων, ένα εργαλείο ανάλυσης συναισθήματος μπορεί να ερμηνεύσει τι λένε οι καταναλωτές, πώς το λένε και ποια είναι η πραγματική τους πρόθεση - τόσο από την οπτική του κάθε ατόμου όσο και από την οπτική του ευρύτερου κοινού.

Μέσα στο πλαίσιο της εξόρυξης κειμένου, η ανάλυση συναισθήματος συνήθως χρησιμοποιείται για τον προσδιορισμό των σχολίων των καταναλωτών σε διάφορα μέσα και πλατφόρμες, όπως κριτικές, έρευνες, ιστοσελίδες και κοινωνικά δίκτυα. Καθώς η γλώσσα εξελίσσεται, γίνεται όλο και πιο προκλητικό να κατανοήσουμε τι εννοούν οι άνθρωποι μέσω αυτών των καναλιών, και η προκαθορισμένη επιλογή λεξιλογίου μπορεί να οδηγήσει σε ανακριβείς ερμηνείες.

Ορισμένοι τύποι ανάλυσης συναισθήματος περιλαμβάνουν τους παρακάτω<sup>1,2</sup>:

- *Πτυχολογική*—Αναγνώριση συγκεκριμένων πτυχών που συζητούν οι καταναλωτές, όπως οι τιμές των προϊόντων στις ιστοσελίδες κριτικών, καθώς και οι αντιδράσεις μεμονωμένων καταναλωτών.
- *Ανίχνευση συναισθημάτων*—Αναγνώριση συναισθημάτων που συνδέονται με συγκεκριμένες λέξεις.
- *Λεπτομερής*—Ανάλυση του συναισθήματος σε διάφορες κατηγορίες πολικότητας (πολύ θετικό, θετικό, ουδέτερο, αρνητικό ή πολύ αρνητικό) για να διευκολυνθεί η κατανόηση των απόψεων των καταναλωτών σε βάθος.
- *Ανάλυση πρόθεσης*—Αναγνώριση της πρόθεσης των καταναλωτών, προκειμένου να γίνει κατανοητό εάν πρόκειται να αγοράσουν ή να διεξάγουν έρευνα.



**Εικόνα 2-1: Παράδειγμα Ανάλυσης Συναισθήματος, (Πηγή: <https://www.gosmar.eu/machinelearning/2020/08/23/recurrent-neural-networks-for-sentiment-analysis/>)**

Παραδοσιακά, οι επιχειρήσεις χρησιμοποιούσαν ερωτηματολόγια και έρευνες για να αξιολογήσουν τη γνώμη των πελατών τους. Για παράδειγμα, η έρευνα Net Promoter Score (NPS) συλλέγει και αξιολογεί πληροφορίες για την προθυμία των πελατών να συστήσουν μια επιχείρηση. Παρόλο που αυτή η προσέγγιση είναι χρήσιμη, μπορεί να περιορίζεται στην παροχή επιπέδου λεπτομερούς πληροφόρησης σχετικά με τις εμπειρίες των πελατών, όπως αυτές που σχετίζονται με την αγορά στα ψηφιακά κανάλια σας.

<sup>1</sup> <https://dynamics.microsoft.com/el-gr/ai/customer-insights/what-is-sentiment-analysis/>

<sup>2</sup> [https://en.wikipedia.org/wiki/Sentiment\\_analysis](https://en.wikipedia.org/wiki/Sentiment_analysis)

Ωστόσο, η ανάλυση συναισθήματος μπορεί να γεφυρώσει αυτό το χάσμα. Μέσω της παρακολούθησης, του εντοπισμού και της εξαγωγής απόψεων και συναισθημάτων από το κείμενο, η ανάλυση συναισθήματος μπορεί να αποκαλύψει το νόημα που κρύβεται πίσω από κάθε σχόλιο, "like" σε κοινωνικά μέσα, ιδέα, παράπονο και ερώτημα. Αυτό θα βοηθήσει τις επιχειρήσεις να ανταποκριθούν ευέλικτα στις συνεχώς εξελισσόμενες ανάγκες των πελατών τους.

Αναλύοντας τα συλλεγόμενα δεδομένα, μπορεί να δημιουργηθεί μια σύνοψη των αντιδράσεων κάθε πελάτη καθώς και οποιαδήποτε πρόσθετη σχόλια που θα μπορούσαν να συμβάλουν στην κατανόηση της αντίληψης του κοινού για το προϊόν ή την επιχείρηση. Όταν αυτά τα δεδομένα ταξινομηθούν σε θετικά, ουδέτερα ή αρνητικά συναισθήματα, οι επιχειρήσεις είναι σε θέση να διακρίνουν τα κίνητρα που οδήγησαν τον πελάτη να εκφράσει μια συγκεκριμένη άποψη, αποκαλύπτοντας τα συναισθήματα που διατυπώνει για ένα συγκεκριμένο θέμα.

Έπειτα, αυτές οι απόψεις ταξινομούνται ως άμεσες ("Αυτό το προϊόν είναι το καλύτερο που έχω χρησιμοποιήσει ποτέ!") ή συγκριτικές ("Το προϊόν A ενσωματώνεται καλύτερα με τη συσκευή μου από το προϊόν B."). Αν και αυτές συνήθως είναι εύκολες στην ερμηνεία, είναι σημαντικό να σημειωθεί ότι ορισμένες από αυτές μπορεί να απαιτούν περαιτέρω ανάλυση. Κατατάσσονται ως σιωπηρές ("Η επιχείρηση γνωρίζει τι πρέπει να κάνει για να βελτιώσει αυτό το προϊόν.") και ρητές ("Η λειτουργία A είναι εύκολη στη χρήση."), καθώς και ακολουθίες λέξεων που είναι θετικές αλλά περιέχουν αρνητικές λέξεις, και μπορεί να είναι δύσκολο να αναλυθούν, με αποτέλεσμα να απαιτηθεί κάποια μη αυτόματη αναθεώρηση ή προσαρμογές στα χρησιμοποιούμενα μοντέλα συναισθημάτων.

## 2.1. Γιατί είναι χρήσιμη η Αναγνώριση Συναισθημάτων

Η ανάλυση συναισθήματος παρέχει τη δυνατότητα κατανόησης της αντίληψης του κοινού, σε αντίθεση με τα μέσα κοινωνικής δικτύωσης που παρέχουν μόνο μια επιφανειακή εικόνα. Για παράδειγμα, ενώ τα retweets στο Twitter μπορεί να φαίνονται θετικά, η ανάλυση συναισθήματος μπορεί να αποκαλύψει αρνητικά σχόλια που αντισταθμίζουν αυτήν τη θετική ανταπόκριση, παρέχοντας μια πιο ακριβή εικόνα της συνολικής αντίδρασης. Επιπλέον, η ανάλυση συναισθήματος μπορεί να αξιοποιήσει τις εσωτερικές πηγές δεδομένων των επιχειρήσεων για την εξαγωγή πολύτιμων πληροφοριών για τους καταναλωτές. Αυτές οι πληροφορίες παρέχουν μια ολοκληρωμένη εικόνα των απόψεων των πελατών και του τρόπου ανταπόκρισής τους. Άλλα οφέλη της ανάλυσης συναισθήματος περιλαμβάνουν:

- Τη χρήση της ως κρίσιμου σημείου για τον εντοπισμό συναισθημάτων πάνω σε ένα συγκεκριμένο θέμα.
- Την εξοικονόμηση χρόνου και προσπάθειας από την αυτοματοποίηση της διαδικασίας εξαγωγής συναισθημάτων.
- Την αξιοποίηση της προσαρμοστικής μάθησης, η οποία επιτρέπει στις επιχειρήσεις να βελτιστοποιούν, να αντιμετωπίζουν προβλήματα και να ανανεώνουν τακτικά τις προβλέψεις.
- Την ταχεία επεξεργασία μεγάλων, μη δομημένων ποσοτήτων δεδομένων για ανάλυση και παροχή πληροφοριών σε πραγματικό χρόνο.

## 2.2. Πώς λειτουργεί η Αναγνώριση Συναισθημάτων

Η ανάλυση συναισθήματος χρησιμοποιεί διάφορες τεχνολογίες για να συγκεντρώσει όλες τις λέξεις και να τις ενοποιήσει σε ένα ενιαίο αντικείμενο με δυνατότητα δράσης. Η διαδικασία ανάλυσης συναισθήματος περιλαμβάνει τέσσερα βήματα:

1. Ανάλυση του κειμένου σε στοιχεία όπως προτάσεις, φράσεις, διακριτικά και μέρη του λόγου.
2. Καθορισμός της φράσης και των συστατικών της.
3. Ανάθεση βαθμολογίας συναισθημάτων σε κάθε φράση με χρήση θετικών ή αρνητικών σημείων.
4. Συνδυασμός των βαθμολογιών για την τελική ανάλυση συναισθημάτων.

Μέσω της καταγραφής περιγραφικών λέξεων και φράσεων και της αντιστοίχισής τους με μια κλίμακα συναισθημάτων, δημιουργείται μια βιβλιοθήκη συναισθημάτων. Μέσω αυτού του συστήματος αξιολόγησης, η ομάδα που υλοποιεί το σύστημα ανάλυσης συναισθημάτων αποφασίζει τη δύναμη και την πολικότητα κάθε λέξης, σημειώνοντας εάν αυτή είναι θετική, αρνητική ή ουδέτερη. Οι πολύγλωσσες μηχανές ανάλυσης συναισθήματος πρέπει επίσης να διατηρούν ξεχωριστές βιβλιοθήκες για κάθε υποστηριζόμενη γλώσσα, ενημερώνοντας τις συνεχώς με νέες φράσεις και απομάκρυνση μη σχετικών όρων.

Η ανάλυση συναισθήματος μπορεί να κατηγοριοποιηθεί σε τρεις διαφορετικές προσεγγίσεις<sup>3</sup>:

- **Βασισμένη σε κανόνες:** Η πιο απλή μορφή ανάλυσης συναισθήματος χρησιμοποιεί λεξικά για να εξετάσει λέξεις και φράσεις και να προσδιορίσει τα συναισθήματά τους. Αυτή η προσέγγιση λειτουργεί καλά με άμεσες και σαφείς απόψεις. Αν και το σύστημα αυτό είναι γρήγορο και εύκολο στη χρήση, σπάνια εξετάζει τον τρόπο με τον οποίο οι λέξεις συνδυάζονται μεταξύ τους σε μια σύνθετη φράση. Θα πρέπει να δημιουργηθούν και να προστεθούν στο σύστημα λεκτικοί κανόνες για συγκριτικές απόψεις, καθώς αυτή η προσέγγιση δυσκολεύεται να κατανοήσει τις σιωπηρές απόψεις. Η ανάλυση συναισθήματος βασισμένη σε κανόνες δεν κάνει χρήση αλγορίθμων μηχανικής μάθησης. Θα εξετάσουμε την κατηγορία αυτή στην υποενότητα 2.2.1.
- **Αυτοματοποιημένη:** Αξιοποιεί στατιστικές, φυσική γλώσσα και μηχανική μάθηση για την ανίχνευση συναισθημάτων. Το σύστημα εκπαιδεύεται να αντιστοιχίζει τις εισόδους με τις αντίστοιχες εξόδους, δηλαδή το κείμενο των πελατών με την πολικότητα του. Οι μηχανές ταξινομούνται βάσει των δεδομένων εισόδου και μπορούν να προσαρμοστούν με τον χρόνο μόλις εκπαιδευτούν. Μπορεί να δοκιμαστεί με επιπλέον δεδομένα για καλύτερες προβλέψεις. Η αυτοματοποιημένη ανάλυση συναισθήματος κάνει χρήση αλγορίθμων μηχανικής μάθησης. Θα εξετάσουμε την κατηγορία αυτή στην υποενότητα 2.2.2.
- **Υβριδική:** Η συνδυασμένη προσέγγιση που συνδυάζει συστήματα βασισμένα σε κανόνες και αυτοματοποιημένα συστήματα επιτρέπει να αποκτηθεί η ακρίβεια που απαιτείται για την πραγματική κατανόηση των καταναλωτών. Αυτό είναι το *ισχυρότερο σύστημα με τη μεγαλύτερη ακρίβεια*, καθώς συνδυάζει τις συναισθηματικές πληροφορίες που συλλέγονται από λεξικά και μπορούν να προσαρμοστούν με τον χρόνο.

### 2.2.1. Αναγνώριση Συναισθημάτων χωρίς τη χρήση Μηχανικής Μάθησης

Η αναγνώριση συναισθημάτων χωρίς τη χρήση μηχανικής μάθησης αναφέρεται σε μια προσέγγιση που βασίζεται σε κανόνες και όχι σε στατιστικές τεχνικές μάθησης. Αυτή η

---

<sup>3</sup> <https://dynamics.microsoft.com/el-gr/ai/customer-insights/what-is-sentiment-analysis/>

προσέγγιση χρησιμοποιεί συγκεκριμένους κανόνες ή λεξικά για να αναγνωρίσει συναισθήματα σε κείμενο.

Οι κανόνες περιέχουν λεξικά με λέξεις και φράσεις που σχετίζονται με συγκεκριμένα συναισθήματα, όπως θετικά, αρνητικά ή ουδέτερα. Ο αναλυτής χρησιμοποιεί αυτούς τους κανόνες για να αξιολογήσει τις λέξεις και τις φράσεις που περιέχονται στο κείμενο και να τα αναθέσει σε συναισθηματικές κατηγορίες.

Για παράδειγμα, ένα απλό λεξικό μπορεί να περιλαμβάνει τη λέξη "χαρούμενος" στην κατηγορία των θετικών συναισθημάτων και τη λέξη "θλίψη" στην κατηγορία των αρνητικών συναισθημάτων. Ο αναλυτής θα αναγνωρίσει αυτές τις λέξεις στο κείμενο και θα τις αξιολογήσει ανάλογα.

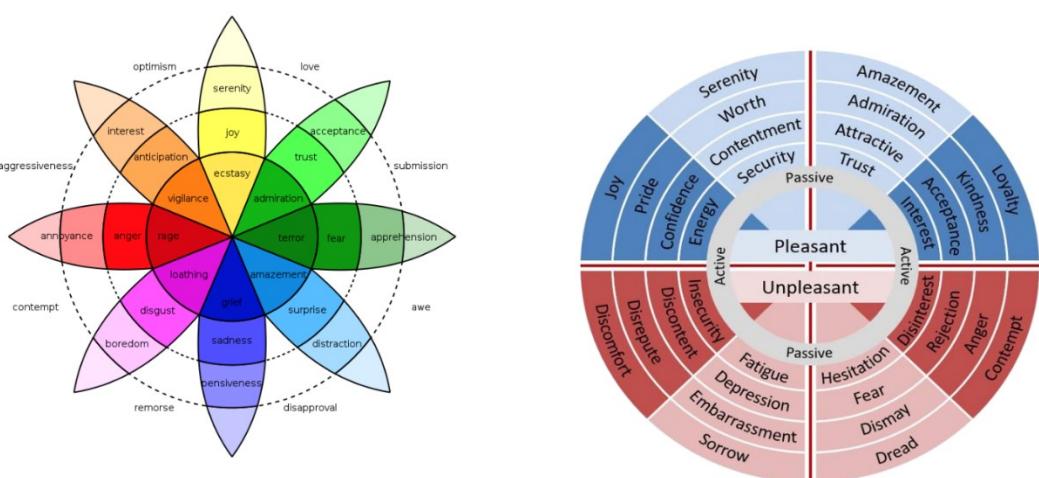
Η αναγνώριση συναισθημάτων με αυτή την προσέγγιση είναι πιο απλή και γρήγορη σε σύγκριση με τις μεθόδους μηχανικής μάθησης, αλλά παρουσιάζει ορισμένους περιορισμούς. Οι κανόνες είναι στατικοί και δεν μπορούν να προσαρμοστούν αυτόματα σε νέα δεδομένα ή σε αλλαγές στον τρόπο έκφρασης των συναισθημάτων. Επίσης, δυσκολεύεται να αντιληφθεί τις σιωπηρές απόψεις ή τις συναισθηματικές υπονοούμενες που μπορεί να περιέχει το κείμενο.

Γενικά, η αναγνώριση συναισθημάτων χωρίς τη χρήση μηχανικής μάθησης είναι μια απλή προσέγγιση που μπορεί να χρησιμοποιηθεί για γρήγορες εκτιμήσεις συναισθημάτων, αλλά δεν παρέχει την ίδια ακρίβεια και ευελιξία με τις πιο σύνθετες μεθόδους που βασίζονται στη μηχανική μάθηση.

### 2.2.1.1. Η ρόδα του Plutchik<sup>4</sup>

Η ρόδα του Plutchik είναι ένα μοντέλο που αναπτύχθηκε από τον ψυχολόγο Robert Plutchik για να αναπαραστήσει τα συναισθήματα και τις σχέσεις μεταξύ τους. Αυτό το μοντέλο παρουσιάζεται ως ένας κύκλος ή ρόδα με οκτώ βασικές συναισθηματικές κατηγορίες που διαιρούνται σε αντιθέτες ζεύγη.

Οι βασικές συναισθηματικές κατηγορίες που παρουσιάζονται στη ρόδα του Plutchik, όπως αυτό παρουσιάζεται στην Εικόνα 2-2 είναι ο ενθουσιασμός-αηδία, η οργή-φόβος, ο προσμονή-έκπληξη, η θλίψη-αγανάκτηση, ο αποτυχία-πείνα, ο ήθος-εκτιμηση, η ήρεμη-αγωνία και η επιπόλαιη-νοσταλγία.



**Εικόνα 2-2: Η ρόδα του Plutchik.** (Πηγή: <https://martecgroup.com/using-plutchiks-wheel-of-emotions-in-market-research/>)

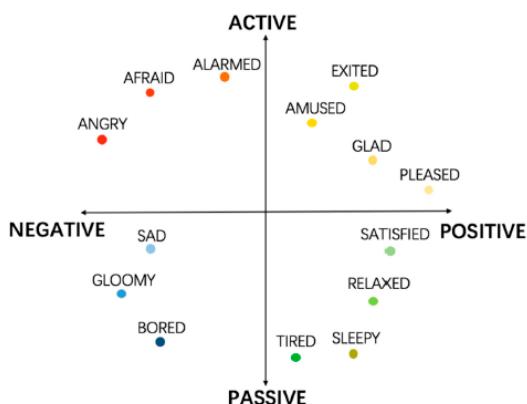
<sup>4</sup> <https://www.6seconds.org/2022/03/13/plutchik-wheel-emotions/>

Σε αυτό το μοντέλο, κάθε βασική συναισθηματική κατηγορία έχει την αντίστοιχη αντίθετη κατηγορία και τις ενδιάμεσες κατηγορίες μεταξύ τους. Οι ενδιάμεσες κατηγορίες αντιπροσωπεύουν συναισθήματα που είναι συνδυασμός των βασικών συναισθημάτων που βρίσκονται δίπλα τους.

Ο σκοπός της ρόδας του Plutchik είναι να παρουσιάσει την πολυπλοκότητα των συναισθημάτων και να καταδείξει τις συναισθηματικές σχέσεις μεταξύ τους. Αυτό το μοντέλο μπορεί να χρησιμοποιηθεί σε πολλούς τομείς, όπως η ψυχολογία, η κοινωνιολογία, η επικοινωνία και η τέχνη, για να κατανοήσουμε και να αναπαραστήσουμε τις ανθρώπινες συναισθηματικές εκφράσεις.

### 2.2.1.2. Δυαδικό συναισθηματικό μοντέλο<sup>5</sup>

Το Δυαδικό Συναισθηματικό Μοντέλο (Εικόνα 2-3) είναι ένα μοντέλο που περιγράφει τα συναισθήματα με βάση ένα ζεύγος αντιθέτων διαστάσεων. Οι δύο βασικές διαστάσεις που χρησιμοποιούνται σε αυτό το μοντέλο είναι η ευχαρίστηση-δυσαρέσκεια και η ενεργητικότητα-απάθεια. Η ευχαρίστηση-δυσαρέσκεια αντιπροσωπεύει τον βαθμό ευχαρίστησης ή δυσαρέσκειας που αισθανόμαστε, ενώ η ενεργητικότητα-απάθεια αντιπροσωπεύει τον βαθμό δραστηριότητας ή αδράνειας που εκφράζουμε.



Εικόνα 2-3: Δισδιάστατο συναισθηματικό μοντέλο. (Shu et al., 2018)

Αυτό το μοντέλο παρουσιάζει τα συναισθήματα σε έναν δισδιάστατο χώρο, με την ευχαρίστηση-δυσαρέσκεια να αποτελεί τον άξονα X και την ενεργητικότητα-απάθεια να αποτελεί τον άξονα Y. Κάθε συναίσθημα αντιπροσωπεύεται από ένα σημείο στον χώρο αυτό, και οι θέσεις των σημείων αντιπροσωπεύουν την ένταση και τη φύση του συναισθήματος.

### 2.2.1.3. Τρισδιάστατο συναισθηματικό μοντέλο<sup>6</sup>

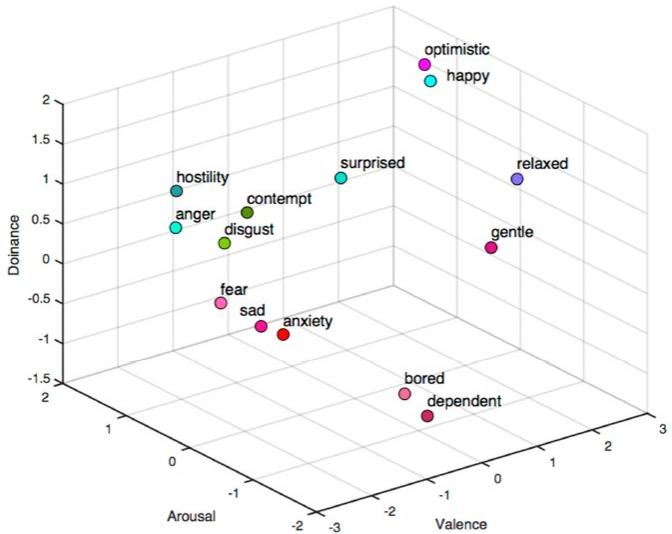
Το Τρισδιάστατο Συναισθηματικό Μοντέλο (Εικόνα 2-4) παρόμοια περιγράφει τα συναισθήματα με βάση τρεις κύριες διαστάσεις. Συνήθως αυτές οι διαστάσεις είναι η ευχαρίστηση-δυσαρέσκεια, η ενεργητικότητα-απάθεια και η αισθητική-λειτουργικότητα. Η αισθητική-λειτουργικότητα αντιπροσωπεύει τον βαθμό αισθητικής απόλαυσης ή αισθητικής απογοήτευσης που συνδέεται με το συναίσθημα.

<sup>5</sup>

[https://dspace.lib.ntua.gr/xmlui/bitstream/handle/123456789/55490/thesis\\_Thodoris\\_Spiliotis.pdf?sequence=1](https://dspace.lib.ntua.gr/xmlui/bitstream/handle/123456789/55490/thesis_Thodoris_Spiliotis.pdf?sequence=1)

<sup>6</sup>

[https://dspace.lib.ntua.gr/xmlui/bitstream/handle/123456789/55490/thesis\\_Thodoris\\_Spiliotis.pdf?sequence=1](https://dspace.lib.ntua.gr/xmlui/bitstream/handle/123456789/55490/thesis_Thodoris_Spiliotis.pdf?sequence=1)



**Εικόνα 2-4: Τρισδιάστατο συναισθηματικό μοντέλο. (Shu et al., 2018)**

Και τα δύο αυτά μοντέλα χρησιμοποιούνται για να κατανοήσουμε και να κατατάξουμε τα συναισθήματα με βάση τις διαστάσεις που παρουσιάζουν. Οι θέσεις των συναισθημάτων στον χώρο των διαστάσεων αντιπροσωπεύουν τη φύση και τη σημασία τους, ενώ η απόσταση μεταξύ των συναισθημάτων αντιπροσωπεύει τον βαθμό της διαφοροποίησής τους. Αυτά τα μοντέλα μπορούν να χρησιμοποιηθούν σε πολλούς τομείς, όπως η ψυχολογία, η κοινωνιολογία, η επικοινωνία και η τέχνη, για να κατανοήσουμε και να αναπαραστήσουμε τις ανθρώπινες συναισθηματικές εκφράσεις.

### 2.2.2. Αναγνώριση Συναισθημάτων με τη χρήση Μηχανικής Μάθησης

Η αναγνώριση συναισθημάτων με τη χρήση μηχανικής μάθησης είναι ένα πεδίο της επιστήμης υπολογιστών που ασχολείται με την ανίχνευση, την κατηγοριοποίηση και την αναγνώριση των συναισθημάτων από δεδομένα όπως κείμενο, φωνή, εικόνες και βίντεο. Η μηχανική μάθηση αποτελεί ένα σημαντικό εργαλείο για την ανάπτυξη αυτών των συστημάτων, καθώς επιτρέπει στους υπολογιστές να μάθουν από δεδομένα και να κατανοήσουν τα χαρακτηριστικά που σχετίζονται με τα συναισθήματα.

Η διαδικασία αναγνώρισης συναισθημάτων με χρήση μηχανικής μάθησης περιλαμβάνει τα εξής βήματα:

1. Συλλογή και προετοιμασία δεδομένων: Αρχικά, συλλέγονται δεδομένα που περιέχουν πληροφορίες για τα συναισθήματα, όπως κείμενα, ηχητικά αρχεία ή εικόνες. Τα δεδομένα αυτά πρέπει να προετοιμαστούν και να μετατραπούν σε μια μορφή που μπορεί να επεξεργαστεί ο αλγόριθμος μηχανικής μάθησης.
2. Εκπαίδευση μοντέλου: Μετά την προετοιμασία των δεδομένων, το μοντέλο μηχανικής μάθησης εκπαιδεύεται χρησιμοποιώντας αυτά τα δεδομένα. Ο αλγόριθμος μηχανικής μάθησης αναλύει τα χαρακτηριστικά των δεδομένων και δημιουργεί ένα μοντέλο που μπορεί να αναγνωρίσει και να κατηγοριοποιήσει τα συναισθήματα.
3. Επικύρωση και αξιολόγηση του μοντέλου: Μετά την εκπαίδευση, το μοντέλο αξιολογείται χρησιμοποιώντας ένα ανεξάρτητο σύνολο δεδομένων που δεν έχει χρησιμοποιηθεί κατά τη διάρκεια της εκπαίδευσης. Αυτό βοηθά να διαπιστωθεί η απόδοση του μοντέλου και να εξαχθούν συμπεράσματα σχετικά με την ακρίβεια και την απόδοσή του.

4. Εφαρμογή και αναγνώριση συναισθημάτων: Μόλις το μοντέλο έχει εκπαιδευτεί και επικυρωθεί, μπορεί να χρησιμοποιηθεί για την αναγνώριση συναισθημάτων σε νέα δεδομένα. Το μοντέλο αξιολογεί τα χαρακτηριστικά των νέων δεδομένων και κατηγοριοποιεί τα συναισθήματα σε προκαθορισμένες κατηγορίες.

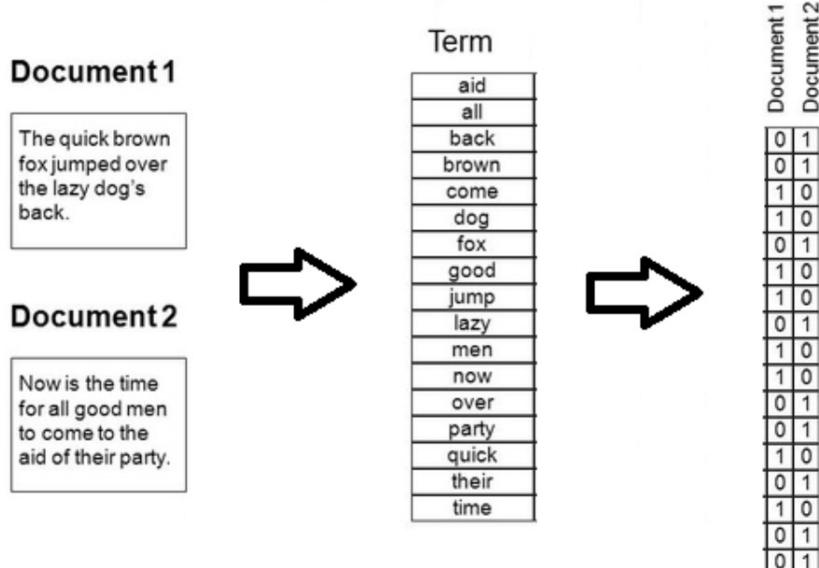
Υπάρχουν κάποιες βασικές κατηγορίες για την αναγνώριση συναισθημάτων με τη χρήση μηχανικής μάθησης. Αυτές οι κατηγορίες βοηθούν στην οργάνωση και την κατηγοριοποίηση των διαφορετικών συναισθημάτων που μπορούν να αναγνωριστούν. Ορισμένες από τις βασικές κατηγορίες είναι οι εξής:

- Θετικά συναισθήματα: Αυτή η κατηγορία περιλαμβάνει συναισθήματα όπως η χαρά, η ικανοποίηση, η ευτυχία και η ενθουσιασμένη κατάσταση. Συνήθως συνδέονται με θετικές εμπειρίες και αντιλήψεις.
- Αρνητικά συναισθήματα: Αυτή η κατηγορία περιλαμβάνει συναισθήματα όπως η θλίψη, η απογοήτευση, ο φόβος και η οργή. Συνήθως συνδέονται με αρνητικές εμπειρίες και δυσάρεστες καταστάσεις.
- Ουδέτερα συναισθήματα: Αυτή η κατηγορία περιλαμβάνει συναισθήματα που δεν εκφράζουν έντονα θετικές ή αρνητικές καταστάσεις. Παραδείγματα μπορεί να είναι η αδιαφορία, η αναισθησία και η αμηχανία.

Αυτές οι κατηγορίες αποτελούν μια γενική προσέγγιση και μπορεί να υπάρχουν περισσότερες συναισθηματικές κατηγορίες ή υποκατηγορίες ανάλογα με το συγκεκριμένο σύστημα αναγνώρισης συναισθημάτων που χρησιμοποιείται. Οι αλγόριθμοι μηχανικής μάθησης επιτρέπουν την αναγνώριση και κατηγοριοποίηση των δεδομένων σε αυτές τις κατηγορίες με βάση τα χαρακτηριστικά που έχουν εκπαιδευτεί να αναγνωρίζουν.

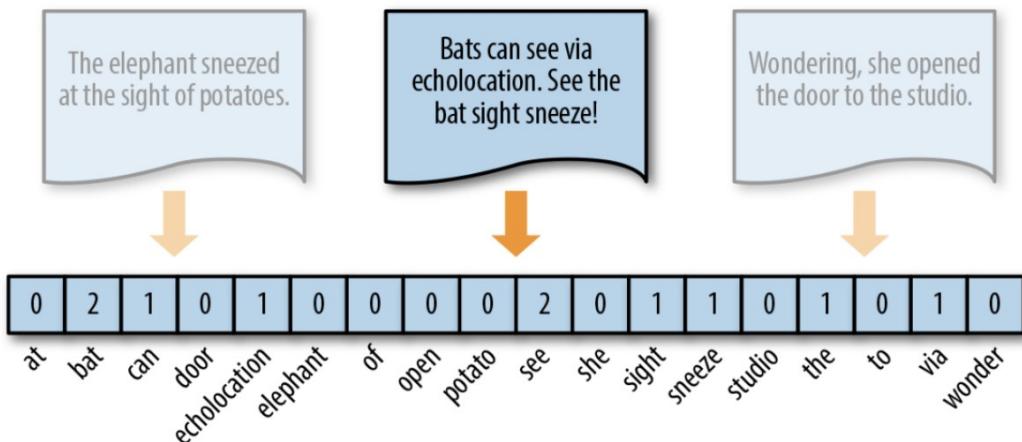
### 2.2.2.1. Σύνολα λέξεων (Bag-of-words)

Τα Σύνολα Λέξεων (Bag-of-words) αναφέρονται σε μια απλή αναπαράσταση κειμένου που αγνοεί τη δομή και τη σειρά των λέξεων και εστιάζει αποκλειστικά στη συχνότητα εμφάνισης των λέξεων σε ένα κείμενο. Στην ουσία, το κείμενο αντιμετωπίζεται ως ένα "σάκο" (bag) που περιέχει όλες τις λέξεις του, αγνοώντας τη σειρά και τη γραμματική δομή.



**Εικόνα 2-5: Παράδειγμα αναπαράστασης Bag-of-words για δυαδικές τιμές<sup>7</sup>.**

Για να δημιουργηθεί ένα Σύνολο Λέξεων, αρχικά το κείμενο διαιρείται σε μικρότερες μονάδες, όπως λέξεις ή ακόμη και γραμματοσειρές. Στη συνέχεια, καταμετρούνται οι εμφανίσεις κάθε λέξης στο κείμενο και αποθηκεύονται ως χαρακτηριστικά. Το τελικό αποτέλεσμα είναι ένα διάνυσμα που αναπαριστά το κείμενο, με κάθε στοιχείο του διανύσματος να αντιστοιχεί σε μια συγκεκριμένη λέξη και την αντίστοιχη συχνότητα εμφάνισής της.



**Εικόνα 2-6: Παράδειγμα αναπαράστασης Bag-of-words για αριθμό εμφανίσεων.**

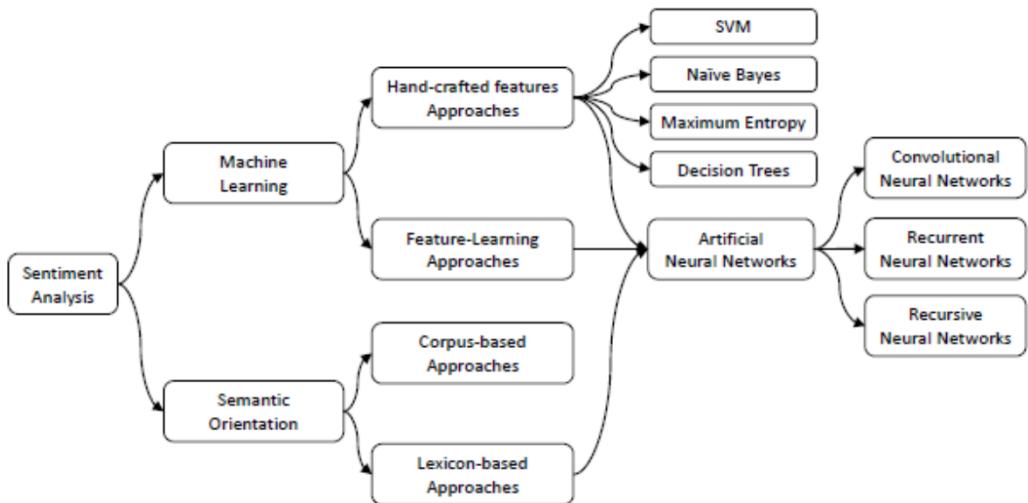
Τα Σύνολα Λέξεων είναι απλά στην υλοποίηση και ευρέως χρησιμοποιούνται σε προβλήματα επεξεργασίας φυσικής γλώσσας και μηχανικής μάθησης. Αν και αποτελούν απλοϊκή αναπαράσταση κειμένου, μπορούν να παρέχουν χρήσιμες πληροφορίες για το περιεχόμενο και το συναισθηματικό περιεχόμενο του κειμένου κατά την εκπαίδευση μοντέλων μηχανικής μάθησης.

### 2.3. Αξιολόγηση μοντέλων Ανάλυσης Συναισθημάτων

Η Εικόνα 2-7 παρουσιάζει μία γενική κατηγοριοποίηση εφαρμόσιμων τεχνικών ανάλυσης συναισθημάτων.

<sup>7</sup>

<https://polynoe.lib.uniwa.gr/xmlui/bitstream/handle/11400/1564/%CE%A7%CE%B1%CF%83%CE% B1%CF%80%CF%8C%CF%80%CE%BF%CF%85%CE%BB%CE%BF%CF%82%20%CE%91%CF %81%CE%B9%CF%83%CF%84%CE%BF%CF%84%CE%AD%CE%BB%CE%B7%CF%82%20- %20%CE%94%CE%B9%CF%80%CE%BB%CF%89%CE%BC%CE%B1%CF%84%CE%B9%CE% BA%CE%AE%20%CE%95%CF%81%CE%B3%CE%B1%CF%83%CE%AF%CE%B1.pdf?sequence =1&isAllowed=y>



**Εικόνα 2-7: Κατηγοριοποίηση εφαρμόσιμων τεχνικών ανά προσέγγιση<sup>8</sup>.**

Ερευνητικές και επιχειρηματικές ομάδες παγκοσμίως ασχολούνται με την ανάλυση συναισθημάτων. Οι επιχειρηματικές ομάδες επικεντρώνονται στην εξόρυξη γνώμης από κριτικές (reviews) και από τα μέσα κοινωνικής δικτύωσης για ένα μεγάλο πλήθος εφαρμογών που εκτείνεται από τη διαφήμιση μέχρι την εξυπηρέτηση πελατών. Από την άλλη μεριά, ο επιστημονικός κλάδος επικεντρώνεται κυρίως στην κατανόηση της δυναμικότητας του συναισθήματος στις ε-κοινωνίες μέσω των τεχνικών της ανάλυσης συναισθημάτων. Το έργο και των δύο δυσχεραίνεται κυρίως από πολιτισμικούς παράγοντες και γλωσσολογικές αποχρώσεις, και σε συνδυασμό με το γεγονός ότι ακόμα και οι άνθρωποι συχνά διαφωνούν μεταξύ τους για το συναισθήμα των κειμένων, αποδεικνύεται τη δυσκολία που αντιμετωπίζει ένας υπολογιστής για την μετατροπή ενός τμήματος γραπτού κειμένου σε κάποιο συναισθήμα. Στον Πίνακα 2.3 παρουσιάζεται η σύγκριση της απόδοσης μεθόδων ανάλυσης συναισθημάτων για διάφορους τομείς και προσεγγίσεις.

**Πίνακας 1: Σύγκριση απόδοσης μεθόδων ανάλυσης συναισθημάτων.**

	Μέθοδος	Σύνολο Δεδομένων	Ακρίβεια
<b>Διαγλωσσική (Cross - lingual)</b>	Ensemble	Amazon	81%
	Co-Train	Amazon, IT168	81.30%
	EWGA	IMDB movie review	> 90%
	CLMM	MPQA, NTCIR, ISI	83.02%
<b>Λεξικογραφική Ανάλυση</b>	Corpus	Product reviews	74%
<b>Διατομεακή</b>	Dictionary	Amazon's Mechanical Turk	
	Active	Book, DVD,	80%

8

<https://polynoe.lib.uniwa.gr/xmlui/bitstream/handle/11400/1564/%CE%A7%CE%B1%CF%83%CE% B1%CF%80%CF%8C%CF%80%CE%BF%CF%85%CE%BB%CE%BF%CF%82%20%CE%91%CF %81%CE%B9%CF%83%CF%84%CE%BF%CF%84%CE%AD%CE%BB%CE%B7%CF%82%20- %20%CE%94%CE%B9%CF%80%CE%BB%CF%89%CE%BC%CE%B1%CF%84%CE%B9%CE% BA%CE%AE%20%CE%95%CF%81%CE%B3%CE%B1%CF%83%CE%AF%CE%B1.pdf?sequence =1&isAllowed=y>

(Cross - domain)	learning	Electronics, Kitchen	(M.O.)
	Thesaurus		
	SFA		
<b>Μηχανική Μάθηση</b>	SVM	Movie reviews	86.4%
	CoTraining	Twitter	82.52%
	SVM		
	Deep Learning	Stanford Sentiment Teebank	80.7%

### 3. Μηχανική Μάθηση

Η **Μηχανική Μάθηση - Machine learning (ML)** είναι ένας τομέας της τεχνητής νοημοσύνης που ασχολείται με τον σχεδιασμό και την ανάπτυξη αλγορίθμων και μοντέλων που επιτρέπουν στις μηχανές (υπολογιστές) να μάθουν από τα δεδομένα και να προβλέπουν ή να λαμβάνουν αποφάσεις χωρίς να προγραμματιστούν αυστηρά για κάθε πιθανή κατάσταση. Ο στόχος είναι να επιτρέπεται στις μηχανές να αντιλαμβάνονται και να ανταποκρίνονται σε δεδομένα, να αναγνωρίζουν πρότυπα και να προβλέψουν τις μελλοντικές τάσεις ή να προτείνουν αποφάσεις βελτιώνοντας την απόδοση των υπολογιστών σε ένα ευρύ σύνολο εργασιών (Wikipedia, 2023).

Ο τρόπος με τον οποίο λειτουργεί η μηχανική μάθηση είναι μέσω της ανάπτυξης μοντέλων με τη χρήση εκπαιδευτικών δεδομένων. Τα εκπαιδευτικά δεδομένα είναι ένα σύνολο δειγμάτων που παρέχονται στο σύστημα, και με αυτά τα δεδομένα το σύστημα "μαθαίνει" και δημιουργεί ένα μοντέλο, το οποίο στη συνέχεια μπορεί να χρησιμοποιηθεί για να προβλέψει ή να αποφασίσει για νέα δεδομένα που δεν έχει "δει" προηγουμένως. Έτσι, οι αλγόριθμοι μηχανικής μάθησης δύνανται να προβλέψουν ή να λάβουν αποφάσεις χωρίς να προγραμματιστούν αυτούσιοι για να το κάνουν (Koza, Bennett, Andre, & Keane, 1996).

Η μηχανική μάθηση έχει εφαρμογές σε πολλούς τομείς, όπως η ιατρική, οι υπηρεσίες τραπεζικής και χρηματοοικονομικής ανάλυσης, οι τηλεπικοινωνίες, η αυτοκινητοβιομηχανία, η ρομποτική, η αναγνώριση φωνής και εικόνας, και πολλές άλλες. Οι αλγόριθμοι μηχανικής μάθησης δύνανται επίσης να χρησιμοποιηθούν σε εφαρμογές στη γεωργία και στην όραση του υπολογιστή (computer vision), ή στο φιλτράρισμα ηλεκτρονικού ταχυδρομείου, περιπτώσεις όπου είναι δύσκολο ή αδύνατο να αναπτυχθούν τυποποιημένοι αλγόριθμοι για την εκτέλεση των αναγκαίων εργασιών (Hu, Niu, Carrasco, Lennox, & Arvin, 2020; Mohsen, Hugh, Tulpan, Sulik, & Eskandari, 2021). Η μηχανική μάθηση συμβάλλει στην αυτοματοποίηση διαδικασιών, τη βελτίωση της ακρίβειας και την ανάπτυξη νέων τεχνολογιών και εφαρμογών.

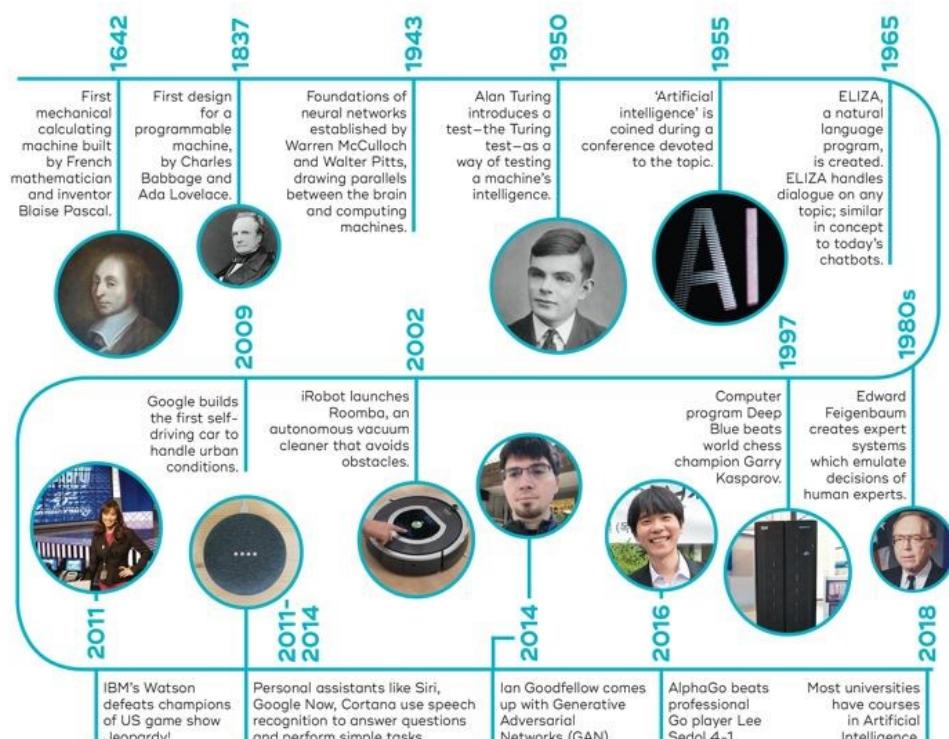
Ένα υποσύνολο της μηχανικής μάθησης σχετίζεται στενά με την υπολογιστική στατιστική, η οποία επικεντρώνεται στην πρόβλεψη χρησιμοποιώντας υπολογιστές, αλλά όχι όλη η μηχανική μάθηση είναι στατιστική μάθηση. Η μελέτη της μαθηματικής βελτιστοποίησης παρέχει μεθόδους, θεωρία και πεδία εφαρμογής στον τομέα της μηχανικής μάθησης. Η εξόρυξη δεδομένων είναι ένας σχετικός τομέας μελέτης, που επικεντρώνεται στην ανακάλυψη και εξερεύνηση δεδομένων μέσω μη επιβλεπόμενης μάθησης (Bishop, 2006). Ορισμένες εφαρμογές της μηχανικής μάθησης χρησιμοποιούν δεδομένα και νευρωνικά δίκτυα με τρόπο που μιμείται τη λειτουργία ενός βιολογικού εγκεφάλου (IBM, 2021). Στην εφαρμογή της σε επιχειρηματικά προβλήματα, η μηχανική μάθηση αναφέρεται επίσης ως προγνωστική αναλυτική.

#### 3.1. Ιστορική αναδρομή

Η ιστορική αναδρομή της μηχανικής μάθησης ξεκινά από τις πρώτες δεκαετίες του 20ού αιώνα, αν και οι ιδέες και οι αλγόριθμοι που τη διέπουν αναπτύχθηκαν περαιτέρω κατά τις δεκαετίες που ακολούθησαν (Wikipedia, 2023). Στη συνέχεια ακολουθούν σημαντικές χρονολογίες από τα κρισιμότερα στάδια στην ιστορία της μηχανικής μάθησης (Wikipedia, 2023):

- ❖ Αρχές του 20ού αιώνα: Οι πρώτες ιδέες για τη μηχανική μάθηση έχουν τις ρίζες τους στον χώρο της στατιστικής και της πιθανότητας. Αλγόριθμοι όπως ο αλγόριθμος Bayes και η γραμμική παλινδρόμηση ήταν από τους πρώτους που χρησιμοποιήθηκαν για την ανάπτυξη μοντέλων πρόβλεψης.

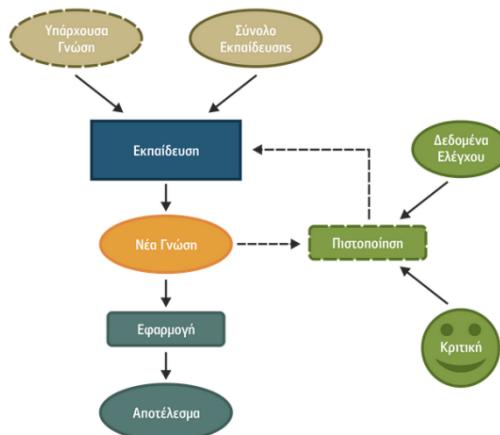
- ❖ Δεκαετία του 1940: Οι έννοιες του νευρωνικού δικτύου και της εκπαίδευσης με βάση την επιβράδυνση (slow learning) εμφανίστηκαν, εμπνευσμένες από τη λειτουργία του ανθρώπινου εγκεφάλου.
- ❖ Δεκαετία του 1950: Ο Alan Turing παρουσίασε την έννοια των "μηχανών που μαθαίνουν", καθώς και τον περίφημο "Τεστ Turing". Επιπλέον, οι ερευνητές Arthur Samuel και Frank Rosenblatt ανέπτυξαν μηχανικά μοντέλα που μπορούσαν να μάθουν να αναγνωρίζουν πρότυπα.
- ❖ Δεκαετία του 1960: Δημιουργήθηκαν οι πρώτοι αλγόριθμοι μάθησης με επίβλεψη, όπως ο αλγόριθμος δέντρου απόφασης ID3, από τον J.R. Quinlan. Επίσης, οι θεωρητικοί υπολογιστές Herbert A. Simon και Allen Newell ανέπτυξαν την έννοια του "μηχανικού μάθησης ως αναζήτησης".
- ❖ Δεκαετίες του 1980-1990: Η μηχανική μάθηση έγινε πιο διαδεδομένη και αναγνωρίστηκε ως ξεχωριστός τομέας. Εμφανίστηκαν νέες τεχνικές, όπως οι μηχανές διανυσμάτων υποστήριξης (SVMs) και οι νευρωνικοί αλγόριθμοι συνελικτικών νευρωνικών δικτύων.
- ❖ Σήμερα: Η μηχανική μάθηση έχει επεκταθεί σε πολλούς τομείς, χάρη στην αύξηση της υπολογιστικής ισχύος και την πρόοδο στη συλλογή και ανάλυση των δεδομένων. Τεχνικές όπως οι νευρωνικοί αλγόριθμοι, οι αλγόριθμοι μάθησης βαθιάς ενίσχυσης και η επεξεργασία φυσικής γλώσσας έχουν αναπτυχθεί σημαντικά, ανοίγοντας νέες προοπτικές σε πολλούς τομείς της κοινωνίας και της τεχνολογίας.



**Εικόνα 3-1: Ιστορική Αναδρομή στη Μηχανική Μάθηση. (Πηγή: <https://tinkeringchild.com/ideas-for-exploring-machine-learning-in-the-primary-years/>)**

### 3.2. Βασικοί αλγόριθμοι Μηχανικής Μάθησης

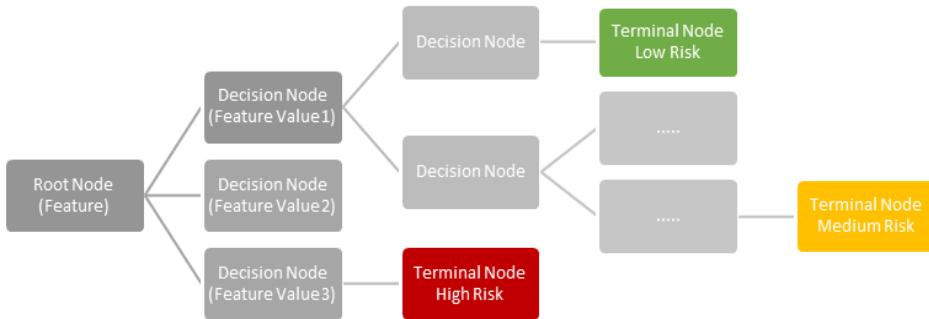
Ένα γενικευμένο μοντέλο Μηχανικής Μάθησης παρουσιάζεται στην Εικόνα 3-2 σε μορφή διαγράμματος ροής. Κάθε διεργασία όπως φαίνεται στο διάγραμμα ροής (για παράδειγμα “Εκπαίδευση”, “Νέα Γνώση”, “Εφαρμογή” κλπ.) δύνανται να πραγματοποιηθεί με μία πληθώρα διαφορετικών αλγορίθμων, τους οποίους θα εξετάσουμε στη συνέχεια της υποενότητας αυτής. Η επιλογή του αλγορίθμου εξαρτάται από το πρόβλημα που επιθυμούμε να επιλύσουμε. Ο κύριος στόχος είναι να καταλήξουμε με το μοντέλο που προσφέρει την καλύτερη ακρίβεια. Συνήθως, για να το καταφέρουμε αυτό, εκπαιδεύουμε το μοντέλο μας με μία σειρά αλγορίθμων και επιλέγουμε τελικά αυτόν με την υψηλότερη απόδοση και ακρίβεια.



Εικόνα 3-2: Ένα γενικευμένο μοντέλο Μηχανικής Μάθησης. (Πηγή: [http://repfiles.kallipos.gr/html\\_books/93/04a-main.html](http://repfiles.kallipos.gr/html_books/93/04a-main.html))

#### 3.2.1. Εκμάθηση με δέντρο απόφασης

Τα Δέντρα Αποφάσεων (Decision Trees - DT) ανήκουν στην κατηγορία των εποπτευόμενων μη παραμετρικών αλγορίθμων μηχανικής μάθησης και χρησιμοποιούνται για ταξινόμηση δεδομένων/παρατηρήσεων. Ο κύριος στόχος του αλγορίθμου DT είναι να δημιουργήσει ένα μοντέλο που μπορεί να προβλέψει την τιμή μιας μεταβλητής στόχου μαθαίνοντας απλούς κανόνες απόφασης που συνάγονται από τα χαρακτηριστικά των δεδομένων (Kotsiantis, 2013). Συγκεκριμένα, το μοντέλο εξάγει χαρακτηριστικά από το σύνολο δεδομένων εκπαίδευσης και οργανώνει ένα δέντρο ταξινόμησης, όπου κάθε κόμβος αντιπροσωπεύει ένα χαρακτηριστικό. Ο αλγόριθμος χωρίζει το σύνολο δεδομένων σε υποσύνολα με βάση τις τιμές των χαρακτηριστικών. Αυτή η διαδικασία επαναλαμβάνεται αναδρομικά για κάθε υποσύνολο, χρησιμοποιώντας μόνο τα δείγματα εκπαίδευσης που ανήκουν στο συγκεκριμένο υποσύνολο. Αν όλα τα δείγματα σε έναν κόμβο έχουν την ίδια ταξινόμηση, τότε η ανάπτυξη του δέντρου διακόπτεται και ο κόμβος αυτός θεωρείται τερματικός. Η κύρια πρόκληση είναι πώς να προσδιορίσετε ποιο χαρακτηριστικό θα διαχωριστεί για να δημιουργηθεί το ταξινομημένο δέντρο. Διάφορες μετρήσεις, όπως ο δείκτης Gini, η Εντροπία και το Κέρδος Πληροφοριών, χρησιμοποιούνται για τον προσδιορισμό του χαρακτηριστικού που θα θεωρηθεί ως ο ριζικός κόμβος, ο οποίος θα διαιρέσει βέλτιστα το σύνολο δεδομένων εκπαίδευσης (Kotsiantis, 2013). Ένα παράδειγμα ταξινόμητη DT για ταξινόμηση ρίσκου απεικονίζεται στην Εικόνα 3-3.



**Εικόνα 3-3: Παράδειγμα εκμάθησης με δέντρο απόφασης.**

Ένα σημαντικό πλεονέκτημα των DT είναι η αποτελεσματική λειτουργία τους ακόμη και με ανεπαρκή δεδομένα, εφόσον ορίζονται κατάλληλοι κανόνες (Kotsiantis, 2013). Επιπλέον, τα DT θεωρούνται πολύτιμα μοντέλα για ταξινόμηση και παρέχουν ευκολία κατανόησης και ερμηνευτικότητας, καθώς είναι δυνατή η οπτικοποίησή τους. Επιπροσθέτως, η προετοιμασία των δεδομένων για τους DT απαιτεί λιγότερο χρόνο σε σύγκριση με άλλους ταξινομητές, όπως η κανονικοποίηση ή η αντιμετώπιση των κενών τιμών. Ωστόσο, όταν τα DT γίνονται μεγάλα, η κατανόησή τους δυσκολεύει και απαιτούν περισσότερα δεδομένα για την εύρεση και επαλήθευση των κανόνων. Τέλος, τα DT μπορεί να είναι ευαίσθητα σε μικρές αλλαγές στις τιμές των χαρακτηριστικών, καθώς αυτές μπορεί να οδηγήσουν σε διαφορετικά συμπεράσματα λόγω της διακριτικότητας των διαμερισμάτων, με αποτέλεσμα τη δημιουργία εντελώς διαφορετικών δέντρων (Kotsiantis, 2013). Αυτό το πρόβλημα μπορεί να αντιμετωπιστεί με την εκπαίδευση πολλών δέντρων σε ένα σύνολο εκπαίδευσης με πλειοψηφία, όπου τα χαρακτηριστικά δειγματίζονται τυχαία με αντικατάσταση ή με τη χρήση DT σε συνδυασμό με άλλους ταξινομητές. Η εκμάθηση συνόλου (ensemble learning) είναι μια πολύ γνωστή στρατηγική για την απόκτηση ακριβών ταξινομητών (Kotsiantis, 2013).

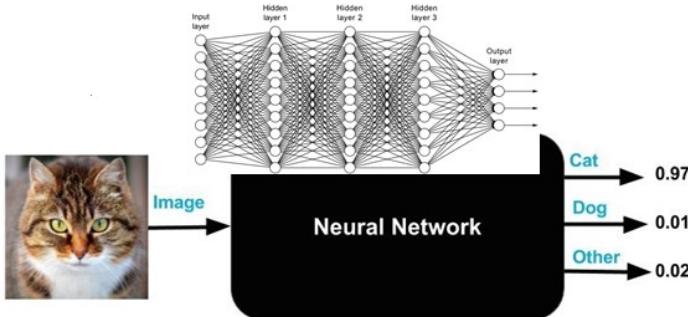
### 3.2.2. Εκμάθηση με κανόνες συσχέτισης

Η εκμάθηση με κανόνες συσχέτισης είναι μια μέθοδος ανευρέσεως αλληλεξαρτήσεων μεταξύ διαφορετικών μεταβλητών σε μεγάλες βάσεις δεδομένων. Η εκμάθηση με κανόνες συσχέτισης αποτελεί κλασσικό παράδειγμα της μάθησης με επίβλεψη καθώς για την εκπαίδευση του αλγορίθμου, είναι απαραίτητη η χρήση ενός αρχικού συνόλου δεδομένων, πάνω στα οποία θα εκπαιδευτεί και θα αναπτυχθεί ο αλγόριθμος.

### 3.2.3. Εκμάθηση με τεχνητά νευρωνικά δίκτυα

Οι αλγόριθμοι εκμάθησης με τεχνητά νευρωνικά δίκτυα, που είναι ευρέως γνωστό απλά ως "νευρωνικό δίκτυο" (Neural Network - NN), είναι ένας αλγόριθμος μάθησης, που εμπνέεται από τη δομή, καθώς και τις λειτουργικές πτυχές των βιολογικών νευρωνικών δικτύων που υπάρχουν στους ζωντανούς οργανισμούς (Acien, Morales, Fierrez, Vera-Rodriguez, & Bartolome, 2020; Kecman, 2001). Η δομή των υπολογισμών βασίζεται σε ένα σύνολο εσωτερικά συνδεδεμένων τεχνητών νευρώνων, οι οποίοι αναλύουν την πληροφορία και εκτελούν υπολογισμούς αλληλεπιδρώντας μεταξύ τους. Τα σύγχρονα νευρωνικά δίκτυα αποτελούν ισχυρά εργαλεία για τη μη γραμμική στατιστική μοντελοποίηση των δεδομένων (Acien et al., 2020; Kecman, 2001). Συνήθως χρησιμοποιούνται για να μοντελοποιήσουν πολύπλοκες σχέσεις μεταξύ των εισερχόμενων και εξερχόμενων δεδομένων, προκειμένου να ανακαλύψουν πρότυπα ή να εντοπίσουν τη στατιστική δομή ενός άγνωστου κοινού

πιθανοτικού μοντέλου μεταξύ των παρατηρούμενων μεταβλητών. (Acien et al., 2020; Kecman, 2001).



**Εικόνα 3-4: Παράδειγμα νευρωνικού δίκτυου.**

Ένα νευρωνικό δίκτυο θεωρείται ένα μαύρο κουτί (black box) με την έννοια ότι, ενώ μπορεί να προσεγγίσει οποιαδήποτε συνάρτηση, η μελέτη της δομής του δεν θα δώσει πληροφορίες σχετικά με τη δομή της συνάρτησης που προσεγγίζεται (Acien et al., 2020; Kecman, 2001). Συνεπώς, από τη μία μπορεί να είναι αποδοτικοί αλγόριθμοι, αλλά από την άλλη δεν είναι επεξηγήσιμοι (explainable), κάτι που είναι πολύ σημαντικό για κρίσιμες εφαρμογές.

### 3.2.4. BERT

Οι αναπαραστάσεις αμφίδρομου κωδικοποιητή από τους μετασχηματιστές (BERT-Bidirectional Encoder Representations from transformers) είναι μια οικογένεια μοντέλων γλώσσας που παρουσιάστηκαν το 2018 από ερευνητές της Google.

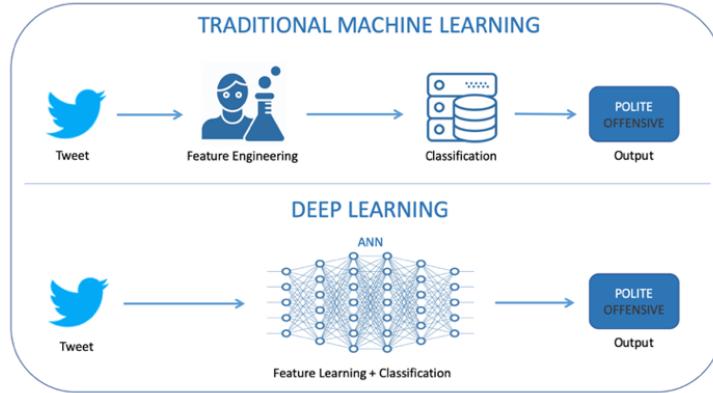
Σε αντίθεση με τα νευρωνικά δίκτυα βαθιάς μάθησης που απαιτούν έναν ευρύ όγκο δεδομένων, το BERT έχει ήδη προεκπαιδευτεί, γεγονός που σημαίνει ότι έχει μάθει τις αναπαραστάσεις των λέξεων και των προτάσεων καθώς και τις εννοιολογικές σχέσεις με τις οποίες συνδέονται.

Πιο συγκεκριμένα, πρόκειται για ένα μοντέλο που εκπαιδεύεται σε αμφίδρομες αναπαραστάσεις κατηγοριοποιημένου κειμένου (labeled text) αναγνωρίζοντας τα συμφραζόμενα μεταξύ αριστερού και δεξιού περιεχομένου. Έτσι το μοντέλο μπορεί να προσαρμοστεί (fine-tune) με την προσθήκη ενός τουλάχιστον επιπέδου (layer) για διαφορετικά tasks όπως question answering (αυτόματη απάντηση σε ερωτήσεις) και language inference. Εφόσον στο σύστημα έχει δοθεί ένα γεγονός ‘premise’ και μία υπόθεση ‘hypothesis’, εξετάζεται αν η υπόθεση είναι αληθής, ψευδής, ή απροσδιόριστη.

Το εν λόγω μοντέλο αντιλαμβάνεται το σκοπό που κρύβεται πίσω από κάθε αναζήτηση. Το γεγονός αυτό φαίνεται απλό για τον αναλυτικό τρόπο σκέψης των ανθρώπων, ωστόσο για μια μηχανή είναι κάτι αρκετά περίπλοκο. Για το λόγο αυτό, απαιτήθηκαν αλλαγές όχι μόνο σε λειτουργικό επίπεδο αλλά και σε επίπεδο μηχανής.

### 3.2.5. Βαθιά μάθηση

Η βασική ιδέα των αλγορίθμων Βαθιάς Μάθησης είναι να μοντελοποιήσουν τον τρόπο με τον οποίο ο ανθρώπινος εγκέφαλος επεξεργάζεται το φως και τον ήχο, μετατρέποντάς τα σε όραση και ακοή. Μερικές από τις επιτυχημένες εφαρμογές της Βαθιάς Μάθησης περιλαμβάνουν τη μηχανική όραση και την αναγνώριση ομιλίας (Hu et al., 2020; Rivera et al., 2020; Sarangi, Sahidullah, & Saha, 2020).



Εικόνα 3-5: Παράδειγμα Μηχανική Μάθησης και Βαθιάς Μάθησης.

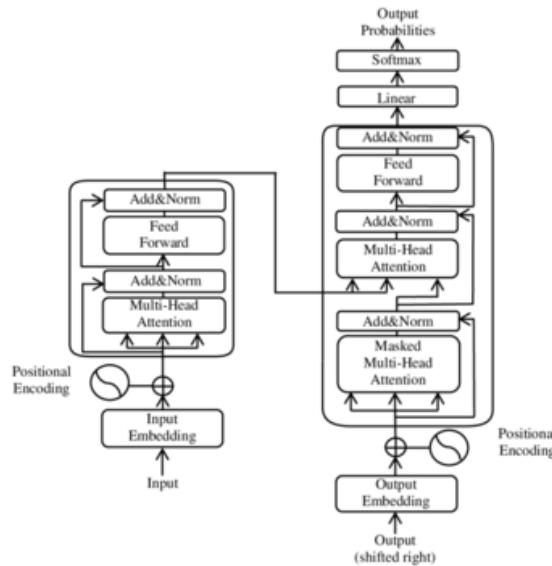
### 3.2.6. Transformers

Ο μετασχηματιστής (Transformer) είναι ένα μοντέλο βαθιάς μάθησης που βασίζεται στον μηχανισμό προσοχής, αξιολογώντας τον αντίκτυπο που έχουν διάφορα τμήματα των εισαγόμενων δεδομένων. Το συγκεκριμένο μοντέλο ξεχώρισε λόγω του ελάχιστου χρόνου εκπαίδευσης που απαιτεί, σε σύγκριση με προηγούμενες επαναλαμβανόμενες νευρωνικές αρχιτεκτονικές, και έχει υιοθετηθεί ευρέως για την εκπαίδευση μεγάλων γλωσσικών μοντέλων σε μεγάλα (γλωσσικά) σύνολα δεδομένων, όπως η Wikipedia Corgus. Αναλυτικότερα, πρόκειται για ένα μοντέλο που λαμβάνει διακριτικά δεδομένα εισόδου (κωδικοποίηση ζεύγους byte), και παράλληλα, μέσω του μηχανισμού προσοχής, ενοποιεί σε κάθε επίπεδο κάθε διακριτικό δεδομένο με άλλα (unmasked) δεδομένα.

Κυκλοφόρησε επίσημα το 2017, ωστόσο ο μηχανισμός προσοχής, τον οποίο υιοθετεί, είχε ήδη προταθεί από το 2014 από τους Bahdanau, Cho και Bengio για αυτόματη μετάφραση.

Αυτή η αρχιτεκτονική έφερε την επανάσταση στον τομέα της τεχνητής νοημοσύνης με την επεξεργασία φυσικής γλώσσας (NLP), ενώ χρησιμοποιείται ακόμη στο computer vision, αλλά και στην επεξεργασία ήχου και σε πολυτροπική επεξεργασία.

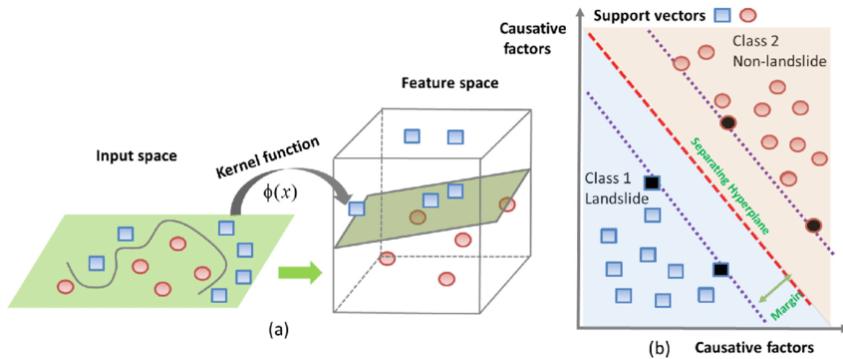
Το μοντέλο Transformer οδήγησε επίσης στην ανάπτυξη προεκπαιδευμένων συστημάτων, όπως το GPT (generative pre-trained transformer) και το BERT (Bidirectional Encoder Representations from transformers).



Εικόνα 3-6: Αρχιτεκτονική Transformer.

### 3.2.7. Εκμάθηση με μηχανές διανυσμάτων υποστήριξης

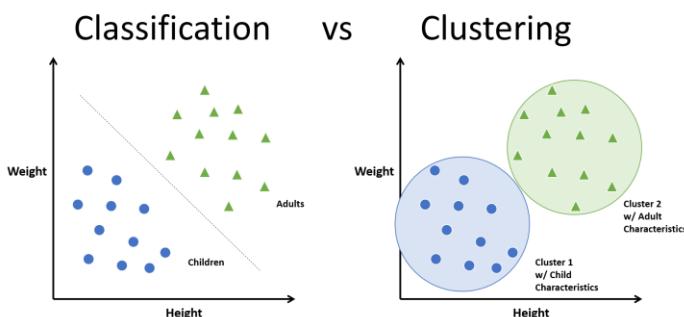
Οι μηχανές διανυσμάτων υποστήριξης είναι ένα σύνολο μεθόδων μάθησης με επίβλεψη που χρησιμοποιούνται για εφαρμογές ταξινόμησης και παλινδρόμησης. Σ' αυτήν την περίπτωση, ένα σύνολο παραδειγμάτων εκπαίδευσης δίνεται ως είσοδο στον αλγόριθμο, ενώ κάθε φορά πρέπει επίσης να δοθεί ως είσοδος στον αλγόριθμο σε ποια από τις δύο κατηγορίες ανήκει το παράδειγμα (ταξινόμηση ή παλινδρόμηση) (Aburomman & Ibne Reaz, 2016; Bishop, 2006; Liu, Ting, & Zhou, 2012). Έτσι, η μηχανή διανυσμάτων υποστήριξης μηχανεύεται ένα μοντέλο που προβλέπει αν ένα νέο παράδειγμα εμπίπτει στην μία κατηγορία ή την άλλη.



**Εικόνα 3-7: Απεικόνιση της αρχής της μάθησης με μηχανές διανυσμάτων υποστήριξης (SVM): (α) Ο χώρος εισόδου αντιστοιχίζεται στον χώρο χαρακτηριστικών με τη βοήθεια μιας συνάρτησης πυρήνα. (β) Διαχωρισμός υπερεπίπεδου και περιθωρίου για ταξινόμηση κατολισθήσεων.**

### 3.2.8. Ομαδοποίηση

Η ομαδοποίηση είναι η διαδικασία μάθησης χωρίς επιτήρηση η οποία εφαρμόζεται ευρέως στη στατιστική ανάλυση δεδομένων. Κατά τη διαδικασία αυτή, ένα σύνολο παρατηρήσεων χωρίζεται σε υποσύνολα, ούτως ώστε οι παρατηρήσεις που ανήκουν στην ίδια ομάδα (*cluster*) να είναι όμοιες μεταξύ τους, σύμφωνα με κάποια προκαθορισμένα κριτήρια. Ταυτόχρονα, οι παρατηρήσεις που ανήκουν σε διαφορετικά υποσύνολα θεωρούνται ανόμοιες. (Iliadis, 2005).



**Εικόνα 3-8: Ταξινόμηση VS Ομαδοποίηση.**

Οι διάφορες τεχνικές κατηγοριοποίησης οδηγούν σε διαφορετικές υποθέσεις σχετικά με τη δομή των δεδομένων. Αυτές οι υποθέσεις συχνά βασίζονται σε μέτρα ομοιότητας και αξιολογούνται, για παράδειγμα, ως προς την εσωτερική συνοχή των μελών του ίδιου *cluster* και τον διαχωρισμό μεταξύ διαφορετικών ομάδων. Μερικές μέθοδοι ομαδοποίησης βασίζονται επίσης στην εκτίμηση της πυκνότητας και της συνεκτικότητας των γραφημάτων.

### 3.2.9. Δίκτυα Bayes

Ένα γραφικό μοντέλο Bayes, γνωστό επίσης ως δίκτυο Bayes ή δίκτυο εμπιστοσύνης, είναι ένα πιθανοθεωρητικό μοντέλο που αναπαριστά τη σχέση μεταξύ μιας συλλογής τυχαίων μεταβλητών μέσω ενός κατευθυνόμενου, ακυκλικού γράφου (Bishop, 2006). Το δίκτυο Bayes μπορεί, για παράδειγμα, να χρησιμοποιηθεί για να μοντελοποιήσει την πιθανοθεωρητική σχέση μεταξύ ασθενειών και συμπτωμάτων. Δεδομένων των συμπτωμάτων, το δίκτυο μπορεί να χρησιμοποιηθεί για να υπολογίσει τις πιθανότητες εμφάνισης διαφόρων ασθενειών.

Ο αλγόριθμος Naive Bayes (NB) είναι ένας εποπτευόμενος αλγόριθμος Μηχανικής Μάθησης που βασίζεται στο θεώρημα του Bayes με μια "αφελή" υπόθεση για την ανεξαρτησία μεταξύ των χαρακτηριστικών, υποθέτοντας δηλαδή ότι κάθε ζεύγος χαρακτηριστικών είναι ανεξάρτητο, δεδομένης της τιμής της μεταβλητής κλάσης, προκειμένου να απλοποιηθεί η διαδικασία μοντελοποίησης (Bishop, 2006). Παρά την αμφιλεγόμενη φύση αυτής της υπόθεσης, ο αλγόριθμος Naive Bayes έχει φανεί γρήγορος και έχει επιδείξει εξαιρετική απόδοση στην πράξη σε πολλούς τομείς. Με δεδομένα τα γεγονότα  $Y$  και  $X$ , όπου  $P(X) \neq 0$ , το θεώρημα του Bayes δηλώνει τα εξής (Bishop, 2006):

$$P(Y|X) = \frac{P(Y)P(X|Y)}{P(X)}$$

Όπου,

$P(Y|X)$ : η υπό όρους πιθανότητα να συμβεί το  $Y$  δεδομένου ότι το  $X$  είναι αληθές,

$P(X|Y)$ : η υπό όρους πιθανότητα να συμβεί το  $X$  δεδομένου ότι το  $Y$  είναι αληθές,

$P(Y)$  : η πιθανότητα να συμβεί το  $Y$  χωρίς καμία προϋπόθεση, και

$P(X)$ : η πιθανότητα να συμβεί το  $X$  χωρίς καμία προϋπόθεση.

Ωστόσο, σε ένα πραγματικό πρόβλημα ταξινόμησης περίπτωσης, μπορεί να υπάρχουν πολλές μεταβλητές  $X$  ανάλογα με τα χαρακτηριστικά των δεδομένων εκπαίδευσης. Ως εκ τούτου, στην κατάσταση στην οποία τα χαρακτηριστικά είναι ανεξάρτητα ή κάτω από αυτήν την υπόθεση, το θεώρημα Bayes επεκτείνεται στον Naïve Bayes (Bishop, 2006):

$$P(Y|X_1, \dots, X_n) = \frac{P(Y)P(X_1, \dots, X_n|Y)}{P(X_1, \dots, X_n)} \quad (1)$$

Με βάση την «αφελή» υπόθεση της ανεξαρτησίας υπό όρους τάξης, τα χαρακτηριστικά είναι υπό όρους ανεξάρτητα το ένα από το άλλο δεδομένης της τάξης, επομένως (Bishop, 2006):

$$P(X_1, \dots, X_n|Y) = P(X_1|Y) \cdots P(X_n|Y) = \prod_{i=1}^n P(X_i|Y) \quad (2)$$

Σύμφωνα με την (1) και (2):

$$P(Y|X_1, \dots, X_n) = \frac{P(Y) \prod_{i=1}^n P(X_i|Y)}{P(X_1, \dots, X_n)} \quad (3)$$

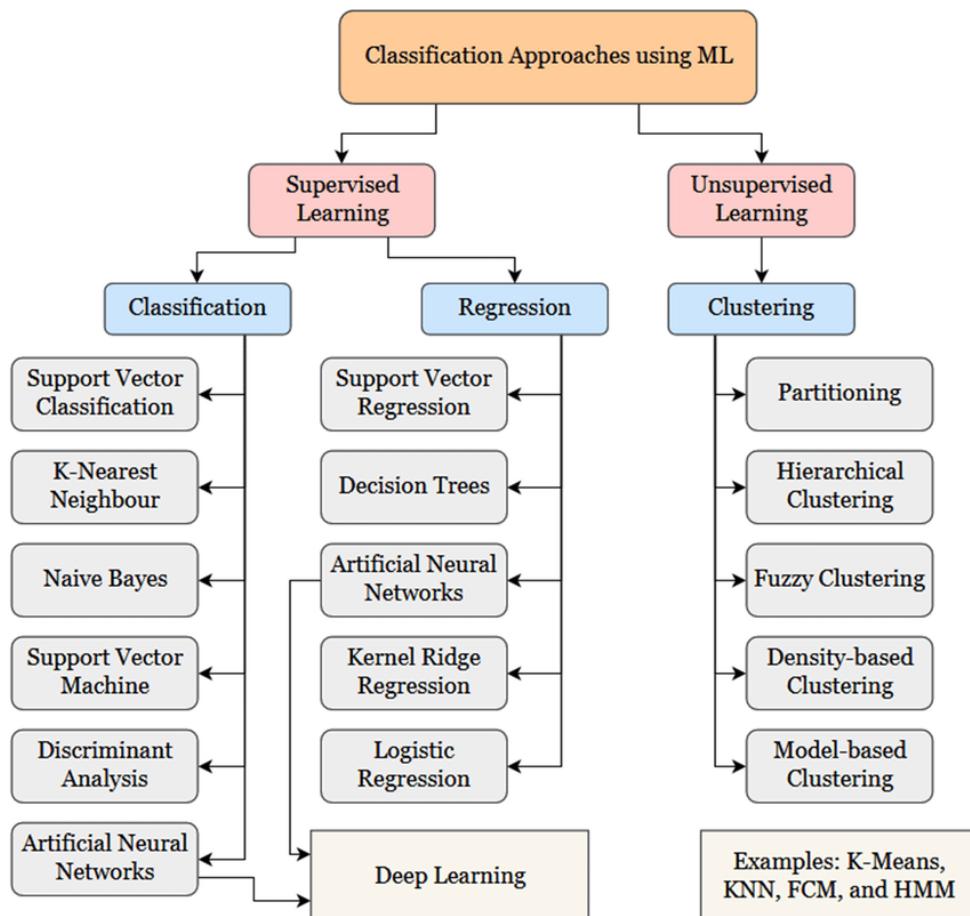
Δεδομένου ότι  $P(X_1, \dots, X_n)$  είναι σταθερή δεδομένης της εισόδου, μπορούμε να χρησιμοποιήσουμε τον ακόλουθο κανόνα ταξινόμησης (Bishop, 2006):

$$P(Y|X_1, \dots, X_n) \propto P(Y) \prod_{i=1}^n P(X_i|Y)$$

$$\hat{Y} = \arg \max_Y P(Y) \prod_{i=1}^n P(X_i|Y)$$

Παρά τις φαινομενικά υπεραπλουστευμένες υποθέσεις τους, οι ταξινομητές NB έχουν ξεπεράσει άλλους πιο εξελιγμένους ταξινομητές σε πολλές εφαρμογές του πραγματικού κόσμου, κυρίως στην ταξινόμηση εγγράφων και στο φιλτράρισμα ανεπιθύμητων μηνυμάτων. Η κύρια πλεονεκτική πτυχή των ταξινομητών NB είναι ότι μπορούν να λειτουργήσουν με μικρό όγκο δεδομένων προεκπαίδευσης για την εκτίμηση των απαραίτητων παραμέτρων. Επιπλέον, οι ταξινομητές NB μπορούν να είναι εξαιρετικά γρήγοροι σε σύγκριση με πιο προηγμένες μεθόδους. Η ανεξαρτησία των κατανομών των χαρακτηριστικών υπό την προϋπόθεση της κλάσης σημαίνει ότι κάθε κατανομή μπορεί να εκτιμηθεί ξεχωριστά ως μονοδιάστατη κατανομή. Αυτό αποβαίνει ως χρήσιμο για την αντιμετώπιση των προβλημάτων που προκύπτουν από τον υψηλό αριθμό διαστάσεων (Bishop, 2006).

Τέλος, ακολουθεί η Εικόνα 3-9, όπου παρουσιάζεται μία ταξινόμηση των αλγορίθμων που χρησιμοποιούνται για ανάπτυξη και εκπαίδευση μοντέλων μηχανική μάθησης με επίβλεψη και μοντέλων μηχανική μάθησης χωρίς επίβλεψη. Όσων αφορά τη μηχανική μάθηση με επίβλεψη, παρουσιάζονται αλγόριθμοι για προβλήματα ταξινόμησης και παλινδρόμησης, ενώ για τη μηχανική μάθηση χωρίς επίβλεψη, παρουσιάζονται οι βασικοί και ευρέως χρησιμοποιούμενοι αλγόριθμοι ομαδοποίησης (Bishop, 2006).



**Εικόνα 3-9: Μηχανική μάθηση με επίβλεψη και χωρίς επίβλεψη - Αλγόριθμοι. (Πηγή: [https://www.researchgate.net/publication/343022075\\_Automatic\\_recognition\\_of\\_handwritten\\_Arabic\\_characters\\_a\\_comprehensive\\_review/figures?lo=1](https://www.researchgate.net/publication/343022075_Automatic_recognition_of_handwritten_Arabic_characters_a_comprehensive_review/figures?lo=1))**

### 3.3. Παραδείγματα εφαρμογών

Οι εφαρμογές της Μηχανικής Μάθησης είναι αμέτρητες, μεταξύ των οποίων:

- Αναγνώριση ομιλίας και γραφικού χαρακτήρα
- Ανάκτηση πληροφορίας
- Βελτιστοποίηση
- Βιοπληροφορική
- Διαδικτυακή Διαφήμιση
- Εντοπισμός Διαδικτυακής απάτης
- Εντοπισμός απάτης πιστωτικής κάρτας
- Επεξεργασία φυσικής γλώσσας
- Ηλεκτρονικά παιχνίδια
- Ιατρική Διάγνωση
- Κατηγοριοποίηση ακολουθιών DNA
- Λογισμικά
- Μαρκετινγκ
- Μετακίνηση Ρομπότ
- Μηχανές αναζήτησης
- Μηχανική αντίληψη
- Οικονομία
- Συναισθηματική υπολογιστική
- Συστήματα σύστασης
- Υπολογιστική ανατομία
- Υπολογιστική όραση- συμπεριλαμβανομένης της αναγνώρισης αντικειμένου
- Χημειοπληροφορική
- Χρηματιστηριακή ανάλυση

Το 2006, η εταιρία ταινιών Netflix διοργάνωσε τον πρώτο διαγωνισμό "Βραβείο Netflix" με σκοπό να βρεθεί ένα πρόγραμμα που θα μπορούσε να προβλέπει τις προτιμήσεις των χρηστών με μεγαλύτερη ακρίβεια και να βελτιώσει τον αλγόριθμο προτεινόμενων ταινιών Cinematch κατά τουλάχιστον 10%.

Μια ομάδα ερευνητών από την AT&T Labs-Research σε συνεργασία με τις ομάδες Big Chaos και Pragmatic Theory ανέπτυξε ένα μοντέλο που κέρδισε το βραβείο ύψους 1 εκατομμυρίου δολαρίων το 2009<sup>9</sup>. Μετά την απονομή του βραβείου, η διοίκηση της Netflix συνειδητοποίησε ότι τα ποσοστά τηλεθέασης δεν ήταν ο καλύτερος δείκτης για τα προτιμήσεις των θεατών ("τα πάντα είναι τηλεθέαση") και προχώρησαν σε αλλαγές στη μηχανή προτάσεων τους<sup>10</sup>.

Το 2010, το περιοδικό The Wall Street αναφέρθηκε στη χρήση της μηχανικής μάθησης από την εταιρεία διαχείρισης χρημάτων Rebellion Research για την πρόβλεψη οικονομικών κινήσεων. Ένα άρθρο περιέγραφε την πρόβλεψη της εταιρείας για την οικονομική κρίση και την οικονομική ανάκαμψη<sup>11</sup>.

---

9

<https://web.archive.org/web/20151110062742/http://www2.research.att.com/~volinsky/netflix/>

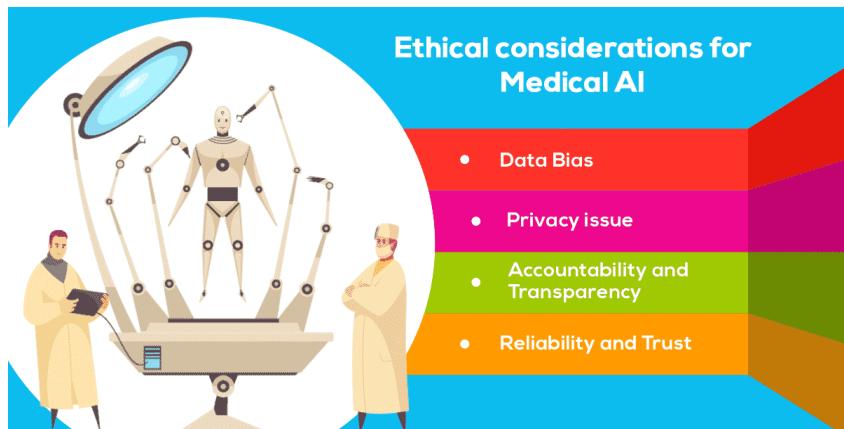
<sup>10</sup> <https://web.archive.org/web/20160531002916/http://techblog.netflix.com/2012/04/netflix-recommendations-beyond-5-stars.html>

<sup>11</sup> <https://www.wsj.com/articles/SB10001424052748703834604575365310813948080>

Το 2014 αναφέρθηκε ότι ένας αλγόριθμος μηχανικής μάθησης χρησιμοποιήθηκε στην Ιστορία της Τέχνης για τη μελέτη πινάκων ζωγραφικής και αποκάλυψε επιρροές μεταξύ καλλιτεχνών που προηγουμένως δεν είχαν ανιχνευθεί<sup>12</sup>. Συνεπώς, καταλαβαίνουμε ότι οι εφαρμογές της μηχανικής μάθησης είναι πολλές και εξερευνούν ένα ευρύ φάσμα τομέων.

### 3.4. Ηθική

Η Μηχανική Μάθηση εισάγει ζητήματα ηθικής άποψης. Αυτό συμβαίνει καθώς τα μοντέλα μηχανικής μάθησης που έχουν εκπαιδευτεί με δεδομένα συλλεγόμενα με προκαταλήψεις δύνανται να εμφανίζουν αυτές τις προκαταλήψεις αργότερα κατά τη χρήση τους σε νέα δεδομένα, ψηφιοποιώντας πολιτιστικές προκαταλήψεις όπως ο ταξικός διαχωρισμός και ο θεσμικός ρατσισμός<sup>13</sup>. Έτσι η υπεύθυνη συλλογή δεδομένων είναι ένα κρίσιμο κομμάτι της μηχανικής μάθησης.



**Εικόνα 3-10: Παράδειγμα ζητημάτων ηθικής σε εφαρμογές μηχανικής μάθησης στο χώρο της υγείας. Συγκεκριμένα: μεροληψία δεδομένων, ζητήματα ιδιωτικότητας, λογοδοσία και διαφάνεια, αξιοπιστία και εμπιστοσύνη.**

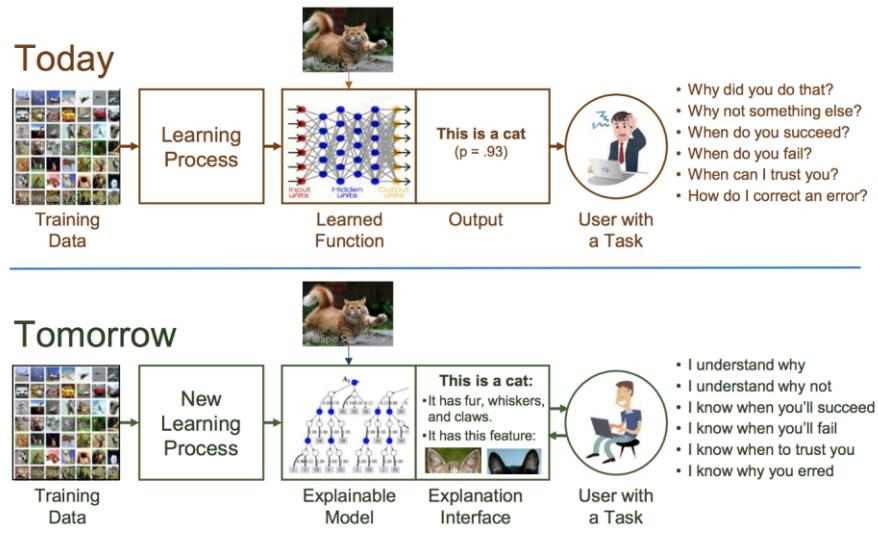
Όταν αναφερόμαστε στην ηθική των αλγορίθμων μηχανικής μάθησης, υπάρχουν 2 έννοιες που εξετάζουμε, οι οποίες μπορεί να χρησιμοποιούνται συχνά εναλλακτικά: η Επεξήγηση και η Ερμηνεία, ωστόσο διαφέρουν σημαντικά. Σύμφωνα με τον KDnuggets (2021), «η ερμηνεία είναι να μπορείς να διακρίνεις τη μηχανική χωρίς απαραίτητα να γνωρίζεις γιατί. Ωστόσο, η επεξήγηση είναι να μπορείς να εξηγήσεις κυριολεκτικά αυτό που συμβαίνει». Πιστεύεται ότι η επεξήγηση αποτελεί ένα τρόπο ενίσχυσης της διαφάνειας των συστημάτων μηχανικής μάθησης, πράγμα το οποίο είναι ιδιαίτερα σημαντικό για τους τελικούς χρήστες. Συχνά η απελευθέρωση (ή ο εξαναγκασμός των οργανισμών να δημοσιοποιήσουν) των δεδομένων στα οποία εκπαιδεύτηκαν τα μοντέλα ή του συνοδευτικού κώδικα αποτελεί πρόκληση λόγω ζητημάτων απορρήτου των χρηστών και κινήτρων για τη διατήρηση του εμπορικού απορρήτου. Επιπλέον, οι τελικοί χρήστες γενικά δεν είναι σε θέση να κατανοήσουν πώς τα ακατέργαστα δεδομένα και αναπτυσσόμενα μοντέλα μηχανικής μάθησης μεταφράζονται σε οφέλη ή βλάβες για τους ίδιους καθώς και για προσωπικές τους πληροφορίες. Έτσι, για την ευρεία αποδοχή και εμπιστοσύνη της μηχανικής μάθησης από τους τελικούς χρήστες, είναι απαραίτητο τα μοντέλα αυτά να παρέχουν διαφάνεια παρέχοντας μια εξήγηση για το πώς

<sup>12</sup> <https://web.archive.org/web/20160604072143/https://medium.com/the-physics-archiv-blog/when-a-machine-learning-algorithm-studied-fine-art-paintings-it-saw-things-art-historians-had-never-b8e4e7bf7d3e>

<sup>13</sup> <https://web.archive.org/web/20160304015020/http://www.nickbostrom.com/ethics/artificial-intelligence.pdf>

έλαβαν μια απόφαση. Η σημασία της επεξήγησης ως έννοιας έχει αντικατοπτριστεί στις νομικές και ηθικές κατευθυντήριες γραμμές για τα μοντέλα μηχανικής μάθησης. Σε περιπτώσεις αυτοματοποιημένης λήψης αποφάσεων, τα άρθρα 13-15 του Ευρωπαϊκού Γενικού Κανονισμού για την Προστασία Δεδομένων (GDPR) απαιτούν τα υποκείμενα των δεδομένων να έχουν πρόσβαση σε «σημαντικές πληροφορίες σχετικά με τη λογική που εμπλέκεται, καθώς και τη σημασία και τις προβλεπόμενες συνέπειες αυτής της επεξεργασίας για το υποκείμενο των δεδομένων» (GDPR). Επιπλέον, εταιρείες τεχνολογίας και ερευνητές μελετούν τις αρχές της τεχνητής νοημοσύνης (AI) στην καθημερινότητα, και εισάγουν τη διαφάνεια ως βασική αξία, συμπεριλαμβανομένων των εννοιών της επεξήγησης, της ερμηνευσιμότητας ή της κατανοητότητας.

Προς αυτή την κατεύθυνση, δεν υπάρχει αμφιβολία ότι η επεξήγηση στη μηχανική μάθηση (ML) και την τεχνητή νοημοσύνη (AI) γίνεται ολοένα και πιο σημαντική, συνθέτοντας μια θεμελιώδη απαίτηση έτσι ώστε οι αναδυόμενες εφαρμογές ML και AI να κερδίσουν την εμπιστοσύνη όλων των εμπλεκόμενων μερών και να φτάσουν τις μέγιστες δυνατότητές τους στην αγορά. Ακριβέστερα, η εξηγήσιμη μηχανική μάθηση επιδιώκει να παρέχει σε διάφορους ενδιαφερόμενους πληροφορίες για τη συμπεριφορά του μοντέλου μέσω βαθμολογιών σπουδαιότητας χαρακτηριστικών, επεξηγήσεων αντιπαραστατικών και δειγμάτων επιρροής, μεταξύ άλλων τεχνικών. Ωστόσο, αυτή η αυξημένη ζήτηση για εξηγήσεις και τον αριθμό των νέων προσεγγίσεων τα τελευταία χρόνια, είναι δύσκολο να βρούμε από πού θα πρέπει να ξεκινήσουμε. Συγκεκριμένα, έχουν μελετηθεί και αναπτυχθεί ποικίλες τεχνικές για την εξήγηση και ερμηνευσιμότητα της απόδοσης των προβλέψεων μοντέλων μηχανικής μάθησης. Αυτό συμβαίνει για διάφορους λόγους, όπως για παράδειγμα, σε ορισμένες εφαρμογές αυτές οι εξηγήσεις μπορεί να απαιτούνται από το νόμο για να διασφαλιστεί ότι οι άνθρωποι θα εμπιστεύονται τέτοιες αποφάσεις που λαμβάνονται από μια «μηχανή». Για να μπορέσουν οι τελικοί χρήστες (άνθρωποι) να είναι σίγουροι και να εμπιστευθούν τα αποτελέσματα, θα πρέπει τα μοντέλα μηχανικής μάθησης να δώσουν τις απαραίτητες επεξηγήσεις και πληροφορίες, δηλαδή γιατί έλαβε ορισμένες αποφάσεις εναντίον άλλων, μαζί με τα τελικά αποτελέσματα. Επιπροσθέτως, η επεξήγηση δύνανται να βοηθήσει στον εντοπισμό των διακρίσεων, του ρατσισμού, και των προκαταλήψεων.



**Εικόνα 3-11: Επεξήγηση στη Μηχανική Μάθηση.**

Γενικώς, υπάρχουν δύο τύποι Επεξήγησιμότητας: α) καθολική (συνολική) συμπεριφορά του μοντέλου και β) τοπική (δηλαδή, επεξήγηση της απόφασης του μοντέλου για κάθε περίπτωση

στα δεδομένα). Ειδικότερα, το πρώτο αφορά τον επεξηγηματικό χαρακτήρα των χαρακτηριστικών του μοντέλου, δηλαδή πόσο σημαντικά και πόσο συμβάλλουν στις προβλέψεις. Στη δεύτερη περίπτωση, η εξήγηση δίνεται για ένα συγκεκριμένο στιγμιότυπο, δηλαδή, τα χαρακτηριστικά που συνέβαλαν περισσότερο στο συγκεκριμένο στιγμιότυπο μπορεί να είναι διαφορετικά για άλλες προβλέψεις. Επιπλέον, ορισμένες προσεγγίσεις εξηγούν τα δεδομένα, άλλες το μοντέλο, ενώ κάποιες μπορεί να είναι καθαρά οπτικές πάνω στα αποτελέσματα. Η επεξηγηση που θα πρέπει να εφαρμόσουμε εξαρτάται από την τελική εφαρμογή που στοχεύουμε να αναπτύξουμε και τι ταιριάζει καλύτερα σε αυτήν.

### 3.5. Λογισμικά που χρησιμοποιούνται ευρέως για Μηχανική Μάθηση

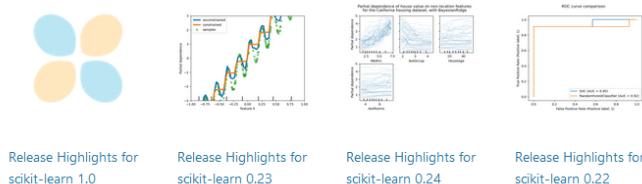
Υπάρχουν αμέτρητα Λογισμικά Μηχανικής Μάθησης που είναι ευρέως διαθέσιμα στην αγορά. Στην υποενότητα αυτή, θα εξετάσουμε τα πιο δημοφιλή μεταξύ αυτών, και στο τέλος θα παραθέσουμε έναν συγκριτικό πίνακα παρουσιάζοντας τα βασικά χαρακτηριστικά τους.

#### 3.5.1. Scikit-learn

Το scikit-learn (<https://scikit-learn.org>) είναι μια λειτουργική μονάδα Python για μηχανική μάθηση που έχει δημιουργηθεί πάνω από το SciPy και διανέμεται με την άδεια 3-Clause BSD license. Το έργο ξεκίνησε το 2007 από τον David Cournapeau ως έργο Google Summer of Code και από τότε πολλοί εθελοντές έχουν συνεισφέρει. Αυτή τη στιγμή συντηρείται από ομάδα εθελοντών. Το Scikit-learn παρέχει συγκεκριμένα μια βιβλιοθήκη για την ανάπτυξη μηχανικής μάθησης σε προγραμματιστική γλώσσα Python (scikit-learn developers, 2023).



Εικόνα 3-12: Περιβάλλον Scikit-learn.

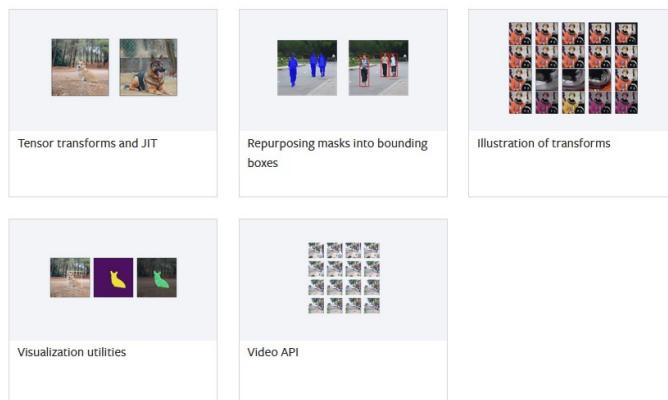


Εικόνα 3-13: Παραδείγματα Scikit-learn. (Πηγή: [https://scikit-learn.org/stable/auto\\_examples/index.html](https://scikit-learn.org/stable/auto_examples/index.html))

Βασικά χαρακτηριστικά του scikit-learn είναι ότι διευκολύνει την εξόρυξη δεδομένων και την ανάλυση δεδομένων, προσφέρει μοντέλα και αλγόριθμους για ταξινόμηση, παλινδρόμηση, ομαδοποίηση, μείωση διαστάσεων, καθώς και προεπεξεργασία των δεδομένων, μία εξίσου σημαντική διαδικασία για την ανάπτυξη μοντέλων μηχανικής μάθησης (Géron, 2019). Ανάμεσα στα πλεονεκτήματά του συμπεριλαμβάνονται ότι είναι ένα αρκετά εύκολο περιβάλλον για χρήση, ιδιαίτερα από χρήστες που δεν έχουν σχέση με τον τομέα αυτό, επιπλέον παρέχεται εύκολα κατανοητή τεκμηρίωση (Géron, 2019). Παράλληλα, οι παράμετροι για οποιονδήποτε συγκεκριμένο αλγόριθμο μπορούν να αλλάξουν κατά την κλήση αντικειμένων, κάτιο το οποίο κάνει τον αλγόριθμο πολύ πιο εύκολο και ευέλικτο. Τέλος, το εργαλείο αυτό είναι διαθέσιμο δωρεάν. (<http://scikit-learn.org/stable/>)

### 3.5.2. PyTorch

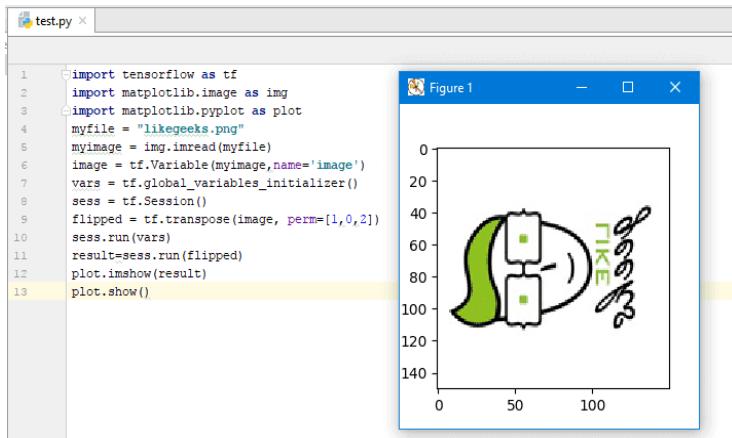
Το PyTorch είναι μια βιβλιοθήκη μηχανικής μάθησης Python που βασίζεται στο Torch. Το Torch αποτελεί μια βιβλιοθήκη υπολογιστικού πλαισίου, γλώσσας σεναρίου και μηχανικής μάθησης που βασίζεται στο λογισμικό Lua (PyTorch, 2023). Βασικά χαρακτηριστικά του είναι ότι βοηθά στη δημιουργία νευρωνικών δικτύων μέσω του Autograd Module. Επιπλέον, παρέχει μια ποικιλία αλγορίθμων βελτιστοποίησης για την κατασκευή νευρωνικών δικτύων. Ένα πολύ σημαίνον πλεονέκτημα είναι ότι το PyTorch είναι δυνατόν να χρησιμοποιηθεί σε πλατφόρμες υπολογιστικού σύννεφου (cloud). Τέλος, προσφέρει κατανεμημένη εκπαίδευση (distributed learning), ενώ παρέχει διάφορα χρήσιμα εργαλεία και βιβλιοθήκες για την ανάπτυξη μοντέλων μηχανικής μάθησης. Ανάμεσα στα πλεονεκτήματά του συμπεριλαμβάνονται ότι διευκολύνει σημαντικά τη δημιουργία υπολογιστικών γραφημάτων και είναι εύκολο στη χρήση του λόγω της υβριδικής του πρόσοψης. Τέλος, το εργαλείο αυτό είναι διαθέσιμο δωρεάν.



**Εικόνα 3-14: Παραδείγματα PyTorch. (Πηγή: [https://pytorch.org/vision/stable/auto\\_examples/index.html](https://pytorch.org/vision/stable/auto_examples/index.html))**

### 3.5.3. TensorFlow

Το TensorFlow αποτελεί μια πλατφόρμα ανοιχτού κώδικα (end-to-end) που χρησιμοποιείται για την ανάπτυξη μοντέλων μηχανικής μάθησης. Διαθέτει ένα επικαιροποιημένο, ολοκληρωμένο και ευέλικτο οικοσύστημα εργαλείων, βιβλιοθηκών και κοινοτικών πόρων που επιτρέπει στους χρήστες να δημιουργούν και να αναπτύσσουν εύκολα εφαρμογές που υποστηρίζονται από μοντέλα μηχανικής μάθησης χρησιμοποιώντας τεχνικές που είναι τελευταία λέξη της τεχνολογίας στον τομέα της μηχανικής μάθησης (Géron, 2019). Το TensorFlow παρέχει μια βιβλιοθήκη JavaScript που βοηθά στη μηχανική εκμάθηση, ενώ η δημιουργία και η εκπαίδευση των τελικών μοντέλων μηχανικής μάθησης γίνεται με APIs (TensorFlow, 2023a).



**Εικόνα 3-15: Παράδειγμα περιβάλλοντος TensorFlow. (Πηγή: <https://dzone.com/articles/tensorflow-simplified-examples?fromrel=true>)**

Βασικά χαρακτηριστικά του είναι ότι βοηθά στην εκπαίδευση και την κατασκευή των μοντέλων μηχανικής μάθησης, παρέχει τη δυνατότητα εκτέλεσης ήδη υπαρχόντων μοντέλων με τη βοήθεια του TensorFlow.js που είναι ένας μετατροπέας μοντέλων (Géron, 2019). Αυτό σημαίνει ότι δεν είναι απαραίτητη η δημιουργία των μοντέλων εξαρχής στο λογισμικό αυτό. Τέλος, είναι εξαιρετικά χρήσιμο για τη δημιουργία νευρωνικών δικτύων.

Συμπερασματικά, το TensorFlow προσφέρει:

1. *Εύκολη κατασκευή μοντέλων μηχανικής μάθησης*

Το TensorFlow ενισχύει την εύκολη, γρήγορη και ευέλικτη δημιουργία και εκπαίδευση μοντέλων μηχανικής μάθησης χρησιμοποιώντας διαισθητικά API υψηλού επιπέδου όπως το Keras με πρόθυμη εκτέλεση, γεγονός που κάνει την άμεση επανάληψη του μοντέλου και τον εύκολο εντοπισμό σφαλμάτων.

2. *Ισχυρή παραγωγή μοντέλων μηχανικής μάθησης οπουδήποτε*

Το TensorFlow ενισχύει την εύκολη, γρήγορη και ευέλικτη εκπαίδευση και ανάπτυξη μοντέλων μηχανικής μάθησης οπουδήποτε: στο cloud, on-prem, στο πρόγραμμα περιήγησης ή σε κινητή συσκευή, ανεξάρτητα από τη γλώσσα προγραμματισμού που η κάθε διαφορετική πλατφόρμα υποστηρίζει.

3. *Δυνατός πειραματισμός για έρευνα*

Το TensorFlow προσφέρει μια απλή και ευέλικτη αρχιτεκτονική για να μεταφέρει τις νέες ιδέες από την ιδέα στον κώδικα, στα μοντέλα τελευταίας τεχνολογίας και στη δημοσίευση επιστημονικών πρότζεκτ πιο γρήγορα και εύκολα.

Ανάμεσα στα πλεονεκτήματά του συμπεριλαμβάνονται ότι παρέχει τη δυνατότητα χρήσης με δύο τρόπους, δηλαδή με ετικέτες σεναρίου (script tags) ή με εγκατάσταση μέσω NPM (Géron, 2019). Παράλληλα, μπορεί ακόμη και να βοηθήσει στην εκτίμηση της ανθρώπινης στάσης. Τέλος, το εργαλείο αυτό είναι διαθέσιμο δωρεάν. Ωστόσο, το λογισμικό αυτό είναι δύσκολο ως προς την εκμάθησή του και τη χρήση του, ιδιαίτερα από χρήστες που δεν είναι εξοικειωμένοι με εργαλεία μηχανική μάθησης γενικότερα (Géron, 2019). Όπως μπορούμε να δούμε και στην Εικόνα 3-15, η χρήση του λογισμικού αυτού απαιτεί γνώσεις προγραμματισμού.

### 3.5.4. Weka

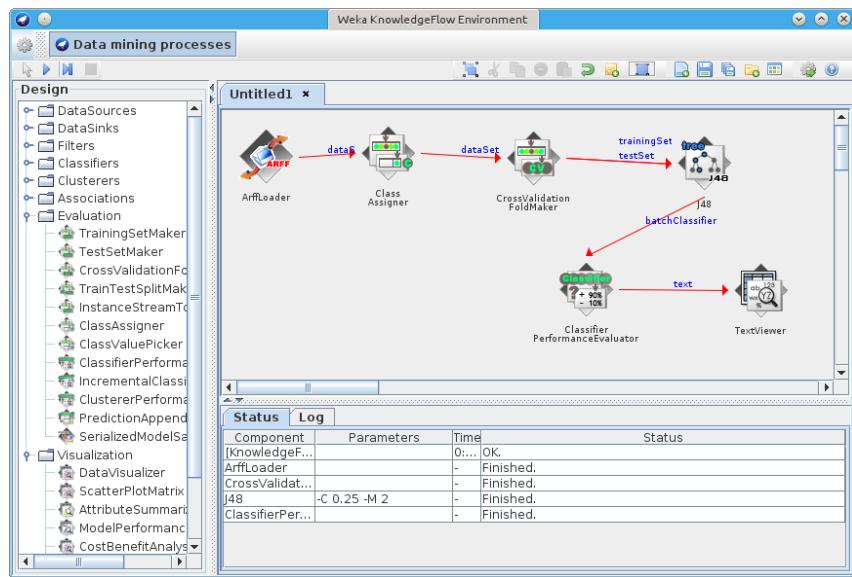
Το λογισμικό Weka παρέχει μια συλλογή εργαλείων οπτικοποίησης και αλγορίθμων για προγνωστική μοντελοποίηση και ανάλυση δεδομένων, μαζί με γραφικές διεπαφές χρήστη για

εύκολη πρόσβαση σε αυτές τις δύο παρεχόμενες λειτουργίες (Witten, Frank, & Hall, 2011). Η αρχική έκδοση του Weka που δεν ήταν Java ήταν μια Tcl/Tk front-end to (mostly third-party) σε αλγόριθμους μοντελοποίησης (κυρίως τρίτων) που εφαρμόστηκαν σε άλλες γλώσσες προγραμματισμού, συν βοηθητικά προγράμματα προεπεξεργασίας δεδομένων σε C και ένα σύστημα βασισμένο σε makefile για την εκτέλεση πειραμάτων μηχανικής εκμάθησης. Η αρχική έκδοση σχεδιάστηκε κυρίως ως εργαλείο για την ανάλυση δεδομένων προερχόμενων από τον γεωργικό τομέα (Holmes, Donkin, & Witten, 1994). Ωστόσο, η πιο πρόσφατη έκδοση του λογισμικού Weka, το Weka 3, για την οποία η ανάπτυξη ξεκίνησε από το 1997, είναι πλήρως βασισμένη στην προγραμματιστική γλώσσα Java, ενώ χρησιμοποιείται πλέον σε πολλές διαφορετικές εφαρμογές συμπεριλαμβανομένων διάφορων τομέων, ιδίως για εκπαιδευτικούς σκοπούς και έρευνα. Τα κύρια πλεονεκτήματα του λογισμικού Weka περιλαμβάνουν τα παρακάτω:

- Ευκολία στη χρήση λόγω των γραφικών διεπαφών χρήστη.
- Δωρεάν διαθεσιμότητα υπό την άδεια GNU (General Public License).
- Παροχή ολοκληρωμένης συλλογής τεχνικών προεπεξεργασίας και μοντελοποίησης δεδομένων.
- Φορητότητα, αφού υλοποιείται πλήρως στη γλώσσα προγραμματισμού Java και έτσι τρέχει σχεδόν σε οποιαδήποτε σύγχρονη υπολογιστική πλατφόρμα.

Παράλληλα, το λογισμικό Weka υποστηρίζει πολλές τυπικές εργασίες εξόρυξης δεδομένων, πιο συγκεκριμένα, προεπεξεργασία δεδομένων, ομαδοποίηση, ταξινόμηση, παλινδρόμηση, οπτικοποίηση και επιλογή χαρακτηριστικών. Η είσοδος στο Weka αναμένεται να μορφοποιηθεί σύμφωνα με τη Μορφή Αρχείου Χαρακτηριστικών Σχέσεων και με το όνομα αρχείου να φέρει την επέκταση .arff. Όλες οι τεχνικές που παρέχει το λογισμικό Weka βασίζονται στην υπόθεση ότι τα δεδομένα που χρησιμοποιούμε είναι διαθέσιμα ως ένα επίπεδο αρχείο ή σχέση, όπου κάθε σημείο δεδομένων περιγράφεται από έναν σταθερό αριθμό χαρακτηριστικών (συνήθως, αριθμητικά ή ονομαστικά χαρακτηριστικά, αλλά υποστηρίζονται και κάποιοι άλλοι τύποι χαρακτηριστικών).

Επιπρόσθετα, το λογισμικό Weka παρέχει τη δυνατότητα πρόσβασης σε βάσεις δεδομένων SQL, ενώ παράλληλα δύνανται να επεξεργαστεί το αποτέλεσμα που επιστρέφεται από ένα ερώτημα βάσης δεδομένων. Το Weka παρέχει τη δυνατότητα πρόσβασης στο DeepLearning4j προσφέροντας τη δυνατότητα ανάπτυξης μοντέλων βαθιάς μάθησης (Holmes et al., 1994). Ωστόσο, δεν είναι ικανό για εξόρυξη δεδομένων πολλαπλών σχέσεων, αλλά υπάρχει ξεχωριστό λογισμικό για τη μετατροπή μιας συλλογής συνδεδεμένων πινάκων βάσεων δεδομένων σε έναν ενιαίο πίνακα που είναι κατάλληλος για επεξεργασία με χρήση του Weka (Reutemann, Pfahringer, & Frank, 2004).



**Εικόνα 3-16: Παράδειγμα περιβάλλοντος Weka.** (Πηγή: <http://open.fracpete.org/2013/09/weka-mooc-has-commenced/>)

Συμπερασματικά, το λογισμικό Weka είναι κατάλληλο για τις παρακάτω εφαρμογές/προβλήματα:

- Προετοιμασία δεδομένων (data preparation)
- Οπτικοποίηση (visualization)
- Ομαδοποίηση (clustering)
- Ταξινόμηση (classification)
- Οπισθοδρόμηση (regression), και
- Κανόνες ένωσης εξόρυξη (association rules mining)

Ανάμεσα στα πλεονεκτήματά του λογισμικού Weka συμπεριλαμβάνονται ότι παρέχει διαδικτυακά μαθήματα για εκπαίδευση, παρέχει εύκολα κατανοητούς αλγόριθμους, είναι ικανοποιητικό και για τους μαθητές. Τέλος, το εργαλείο αυτό είναι διαθέσιμο δωρεάν. Ωστόσο, σημαντικό μειονέκτημα είναι ότι δεν υπάρχουν πολλά έγγραφα και ηλεκτρονική υποστήριξη για το λογισμικό αυτό. Πάντως, είναι αρκετά εύκολο στη δημιουργία και χρήση μοντέλων μηχανικής μάθησης.

### 3.5.5. Google Colab

Το Google Colab είναι ένα λογισμικό στο cloud που υποστηρίζει Python, το οποίο διευκολύνει τη δημιουργία εφαρμογών μηχανικής εκμάθησης χρησιμοποιώντας τις βιβλιοθήκες των PyTorch, Keras, TensorFlow και OpenCV. Βασικά χαρακτηριστικά του λογισμικού αυτού είναι ότι βοηθά στην εκπαίδευση μηχανικής μάθησης, καθώς επίσης ότι βοηθά στην έρευνα μηχανικής μάθησης. Ανάμεσα στα πλεονεκτήματά του συμπεριλαμβάνονται ότι μπορεί να γίνει χρήση του συγκεκριμένου λογισμικού εύκολα και γρήγορα μέσα από το google drive του χρήστη. Τέλος, το εργαλείο αυτό είναι διαθέσιμο δωρεάν.

Πιο συγκεκριμένα, το Colab επιτρέπει σε ένα χρήστη να γράψει και να εκτελέσει Python στο πρόγραμμα περιήγησής του (Google, 2023), με

- Μηδενική διαμόρφωση
- Εύκολη κοινή χρήση
- Δωρεάν πρόσβαση σε GPU

Είτε πρόκειται για έναν φοιτητή, είτε επιστήμονα δεδομένων είτε ερευνητή τεχνητής νοημοσύνης, το Colab μπορεί να διευκολύνει τη δημιουργία μοντέλων μηχανικής μάθησης για ένα ευρύ φάσμα εφαρμογών και προβλημάτων. Η αρχική σελίδα του λογισμικού δεν είναι μια στατική ιστοσελίδα, αλλά ένα διαδραστικό περιβάλλον που ονομάζεται σημειωματάριο Colab το οποίο επιτρέπει στο χρήστη να γράψει και να εκτελέσει κώδικα δημιουργώντας μοντέλα μηχανικής μάθησης.

Για παράδειγμα, παρακάτω είναι ένα κελί κώδικα με ένα σύντομο σενάριο Python που υπολογίζει μια τιμή, την αποθηκεύει σε μια μεταβλητή και εκτυπώνει το αποτέλεσμα:

```
[ ] seconds_in_a_day = 24 * 60 * 60
seconds_in_a_day
86400
```

**Εικόνα 3-17: Παράδειγμα 1 στο περιβάλλον Colab. [38]**

Για να εκτελεστεί ο κώδικας στο παραπάνω κελί, ο χρήστης πρέπει απλά να τον επιλέξει με ένα κλικ και, στη συνέχεια, να πατήσει το κουμπί αναπαραγωγής στα αριστερά του κώδικα ή να χρησιμοποιήσει τη συντόμευση πληκτρολογίου "Command/Ctrl+Enter". Για την επεξεργασία του κώδικα, ο χρήστης πρέπει απλά να κάνει κλικ στο κελί και αυτόματα ξεκινά την επεξεργασία του. Οι μεταβλητές που ο χρήστης ορίζει σε ένα κελί μπορούν αργότερα να χρησιμοποιηθούν σε άλλα κελιά:

```
[ ] seconds_in_a_week = 7 * seconds_in_a_day
seconds_in_a_week
604800
```

**Εικόνα 3-18: Παράδειγμα 2 στο περιβάλλον Colab. [38]**

Τα σημειωματάρια Colab επιτρέπουν στο χρήστη να συνδυάζει διάφορες μορφές δεδομένων και εντελών, συμπεριλαμβανομένου εκτελέσιμου κώδικα και εμπλουτισμένο κείμενο σε ένα μόνο έγγραφο, μαζί με εικόνες, HTML, LaTeX και άλλα. Έτσι, τα σημειωματάρια Colab δημιουργούν ένα εύκολο και ευέλικτο περιβάλλον προσφέροντας καλύτερη εμπειρία του χρήστη (user experience). Τα σημειωματάρια Colab που δημιουργεί ένας χρήστης αποθηκεύονται απευθείας στον λογαριασμό του στο Google Drive. Κατά συνέπεια, είναι πλέον εύκολα και εφικτό ένας χρήστης να μοιραστεί τα σημειωματάρια Colab με συναδέλφους ή φίλους, επιτρέποντάς τους να σχολιάσουν τα σημειωματάριά ή ακόμα και να τα επεξεργαστούν. Τα σημειωματάρια Colab είναι σημειωματάρια Jupyter που φιλοξενούνται από την Colab.

Το Colab παρέχει πλήρη αξιοποίηση των δημοφιλών βιβλιοθηκών Python για την ανάλυση και την οπτικοποίηση των δεδομένων. Για παράδειγμα, το παρακάτω κελί κώδικα χρησιμοποιεί το numpy για τη δημιουργία τυχαίων δεδομένων και χρησιμοποιεί το matplotlib για να τα οπτικοποιήσει. Το Colab κάνει δυνατή την εισαγωγή δεδομένων του χρήστη σε σημειωματάρια Colab από τον λογαριασμό του χρήστη στο Google Drive, μεταξύ άλλων από υπολογιστικά φύλλα, καθώς και από το Github και πολλές άλλες πηγές. Παράλληλα, με το Colab ο χρήστης μπορεί να εισαγάγει ένα σύνολο δεδομένων εικόνας, να εκπαιδεύσει έναν ταξινομητή εικόνας σε αυτό και να αξιολογήσει το μοντέλο, όλα αυτά σε λίγες μόνο γραμμές κώδικα. Τα σημειωματάρια Colab εκτελούν κώδικα στους διακομιστές cloud της Google, πράγμα που σημαίνει ότι αξιοποιούν τη δύναμη του υλικού Google, συμπεριλαμβανομένων των GPU και των TPU, ανεξάρτητα από την ισχύ του μηχανήματος του χρήστη. Το μόνο που πραγματικά

χρειάζεται κάποιος για να δημιουργήσει και να εκπαιδεύσει μοντέλα μηχανικής μάθησης στο λογισμικό Colab είναι ένα πρόγραμμα περιήγησης και σύνδεση στο ίντερνετ.



**Εικόνα 3-19: Παράδειγμα 3 στο περιβάλλον Colab. [38]**

### 3.5.6. Apache Mahout

Το Apache Mahout σχετίζεται άμεσα με το Apache Hadoop, το οποίο αποτελεί ένα πλαίσιο ανοιχτού κώδικα από τον Apache που επιτρέπει την αποθήκευση και επεξεργασία μεγάλων όγκων δεδομένων σε ομάδες υπολογιστών σε κατανεμημένο περιβάλλον χρησιμοποιώντας απλά μοντέλα προγραμματισμού. Το Apache Mahout αποτελεί έργο ανοιχτού κώδικα που χρησιμοποιείται κυρίως για τη δημιουργία κλιμακωτών αλγορίθμων μηχανικής μάθησης. Για να το κάνει αυτό, εφαρμόζει δημοφιλείς τεχνικές μηχανικής εκμάθησης όπως: Ταξινόμηση, Σύσταση, και Ομαδοποίηση.

Το Apache Mahout ξεκίνησε ως υπο-έργο του Apache's Lucene το 2008, ενώ αργότερα το 2010, ο Mahout έγινε έργο κορυφαίου επιπέδου του Apache. Τα πρωτόγονα χαρακτηριστικά του Apache Mahout παρατίθενται παρακάτω:

1. Οι αλγόριθμοι του Mahout λειτουργούν αποδοτικά σε κατανεμημένα περιβάλλοντα, δεδομένου ότι είναι γραμμένοι πάνω από το Hadoop. Το Mahout χρησιμοποιεί τη βιβλιοθήκη Apache Hadoop για αποτελεσματική λειτουργία στο cloud.
2. Το Mahout προσφέρει στον χρήστη ένα έτοιμο προς χρήση πλαίσιο για την εκτέλεση εργασιών εξόρυξης δεδομένων σε μεγάλους όγκους δεδομένων.
3. Το Mahout επιτρέπει τη γρήγορη και αποτελεσματική ανάλυση μεγάλων συνόλων δεδομένων.
4. Περιλαμβάνει πολλές υλοποίησεις ομαδοποίησης με δυνατότητα MapReduce, όπως k-means, fuzzy k-means, Canopy, Dirichlet και Mean-Shift.
5. Υποστηρίζει εφαρμογές ταξινόμησης, όπως Distributed Naive Bayes και Complementary Naive Bayes.
6. Προσφέρει κατανεμημένες δυνατότητες λειτουργιών φυσικής κατάστασης για εξελικτικό προγραμματισμό.
7. Περιλαμβάνει βιβλιοθήκες μητρώων και διανυσμάτων.

Το Mahout χρησιμοποιείται ευρέως σε διάφορες άλλες ήδη ανεπτυγμένες εφαρμογές. Για παράδειγμα:

- Το Twitter χρησιμοποιεί το Mahout για μοντελοποίηση ενδιαφέροντος χρήστη.

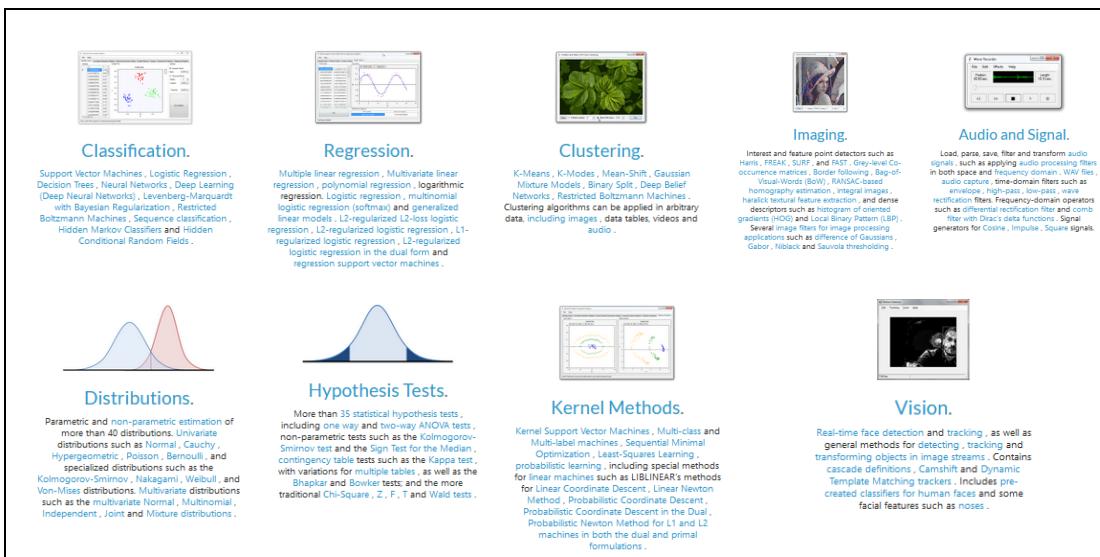
- Εταιρείες όπως οι Adobe, Facebook, LinkedIn, Foursquare, Twitter και Yahoo χρησιμοποιούν το Mahout εσωτερικά για τη δημιουργία μοντέλων μηχανικής μάθησης για τις κύριες λειτουργίες τους.
- Το Yahoo! χρησιμοποιεί το Mahout για εξόρυξη προτύπων.
- Το Foursquare βοηθά το χρήστη να ανακαλύψει μέρη, φαγητό και ψυχαγωγία διαθέσιμα σε μια συγκεκριμένη περιοχή, χρησιμοποιώντας τη μηχανή συστάσεων της Mahout.

### 3.5.7. Accord.NET

Το Accord.NET είναι ένα πλαίσιο για επιστημονικούς υπολογισμούς στο .NET. Ο πηγαίος κώδικας του έργου είναι διαθέσιμος υπό τους όρους της Gnu Lesser Public License, έκδοση 2.1 (Souza, 2017). Το συγκεκριμένο λογισμικό παρέχει ένα ευρύ σύνολο βιβλιοθηκών οι οποίες είναι διαθέσιμες είτε στον πηγαίο κώδικα, είτε μέσω εκτελέσιμων προγραμμάτων εγκατάστασης και πακέτων NuGet. Οι κύριες εφαρμογές που καλύπτει το Accord.NET είναι η μηχανική μάθηση, αριθμητική βελτιστοποίηση, στατιστική, τεχνητά νευρωνικά δίκτυα, αριθμητική γραμμική άλγεβρα, επεξεργασία σήματος και εικόνας, καθώς επίσης και βιβλιοθήκες υποστήριξης (όπως γραφική παράσταση και οπτικοποίηση) (Black Duck Open Hub, 2023). Αρχικά, το λογισμικό αυτό δημιουργήθηκε για να επεκτείνει τις δυνατότητες του AForge.NET Framework, αλλά έκτοτε έχει τελικά ενσωματώσει το λογισμικό AForge.NET και μπορεί να χρησιμοποιηθεί αυτόνομα για τις εφαρμογές που αναφέραμε παραπάνω. Οι νεότερες εκδόσεις του Accord.NET έχουν ενώσει και τα δύο πλαίσια (AForge.NET Framework και AForge.NET), και πλέον είναι γνωστό με το όνομα Accord.NET. Βασικά χαρακτηριστικά του λογισμικού αυτού είναι ότι παρέχει αλγόριθμους για:

- Στατιστική
- Επεξεργασία εικόνας, ήχου και σήματος. Αριθμητική γραμμική άλγεβρα
- Τεχνητά Νευρωνικά Δίκτυα.
- Αριθμητική βελτιστοποίηση
- Παρέχει επίσης υποστήριξη για βιβλιοθήκες σχεδίασης γραφημάτων και οπτικοποίησης.

Ένα από τα βασικά του πλεονεκτήματα είναι ότι οι βιβλιοθήκες του διατίθενται από τον πηγαίο κώδικα και επίσης μέσω του εκτελέσιμου προγράμματος εγκατάστασης και του διαχειριστή πακέτων NuGet. Τέλος, το εργαλείο αυτό είναι διαθέσιμο δωρεάν. Ωστόσο, το Accord.NET υποστηρίζει μόνο .NET, κάτι το οποίο κάνει το λογισμικό αυτό λιγότερο ευέλικτο σε σύγκριση με λογισμικά που παρουσιάστηκαν παραπάνω.



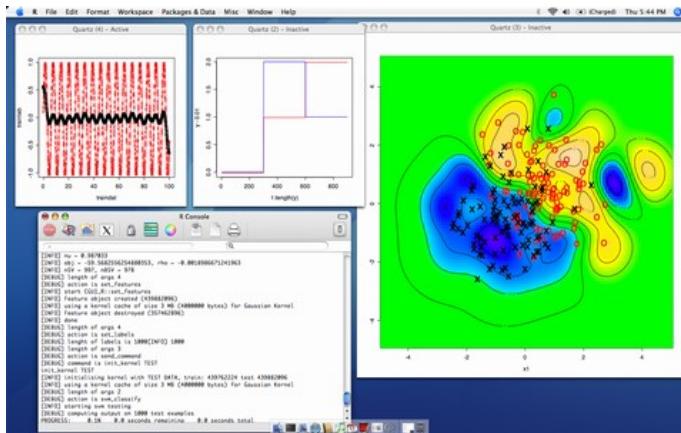
**Εικόνα 3-20: Machine learning made in a minute / Μηχανική μάθηση στο λεπτό / Εφαρμογές Accord.NET**  
 (Πηγή: <http://accord-framework.net/>)

### 3.5.8. Shogun

Το Shogun είναι μια δωρεάν βιβλιοθήκη λογισμικού μηχανικής εκμάθησης ανοιχτού κώδικα σε C++, η οποία προσφέρει πολλούς αλγόριθμους και δομές δεδομένων για προβλήματα μηχανικής μάθησης. Παράλληλα, προσφέρει διεπαφές για Octave, Python, R, Java, Lua, Ruby και C# χρησιμοποιώντας το SWIG (Sonnenburg et al., 2010). Η δωρεάν χρήση του συγκεκριμένου λογισμικού έγκειται σύμφωνα με τους όρους της Γενικής Άδειας Δημόσιας Χρήσης GNU, έκδοση 3 ή μεταγενέστερη.

Το λογισμικό Shogun εστιάζει σε μηχανές πυρήνα (kernel machines), όπως μηχανές υποστήριξης διανυσμάτων για την επίλυση προβλημάτων ταξινόμησης και παλινδρόμησης. Επιπρόσθετα, προσφέρει μια πλήρη εφαρμογή των μοντέλων Hidden Markov. Ο πυρήνας του Shogun είναι γραμμένος σε C++ και προσφέρει διεπαφές για MATLAB, Octave, Python, R, Java, Lua, Ruby και C#. Το συγκεκριμένο λογισμικό βρίσκεται υπό ενεργό ανάπτυξη από το 1999. Σήμερα υπάρχει μια ζωντανή κοινότητα χρηστών σε όλο τον κόσμο που χρησιμοποιεί το Shogun ως βάση για έρευνα και εκπαίδευση. Η ομάδα αυτή συνεισφέρει άμεσα στο βασικό πακέτο που προσφέρει το Shogun (Gashler, 2011). Επί του παρόντος, το Shogun υποστηρίζει τους ακόλουθους αλγόριθμους:

- Υποστήριξη διανυσματικά μηχανήματα
- Αλγόριθμοι μείωσης διαστάσεων, όπως PCA, Kernel PCA, Locally Linear Embedding, Hessian Locally Linear Embedding, Local Tangent Space Alignment, Linear Local Tangent Space Alignment, Kernel Locally Linear Embedding, Kernel Local Tangent Space Isign,p Local Tangent Space Alignment,p. Ιδιοχάρτες Λαπλασίας
- Διαδικτυακοί αλγόριθμοι εκμάθησης όπως SGD-QN, Vowpal Wabbit
- Γραμμική διακριτική ανάλυση
- Αλγόριθμοι ομαδοποίησης: k-means και GMM
- Παλινδρόμηση Kernel Ridge, Υποστήριξη Vector Regression
- K-Κοντινότεροι Γείτονες
- Hidden Markov Models (HMMs)



**Εικόνα 3-21: Στιγμιότυπο οθόνης της εργαλειοθήκης SHOGUN.**

Το λογισμικό Shogun αναπτύχθηκε κατά κύριο λόγο εφαρμογές βιοπληροφορικής, κι έτσι το δυνατό του σημείο είναι ότι παρέχει δυνατότητες επεξεργασίας τεράστιων συνόλων δεδομένων (έως και 10 εκατομμύρια δείγματα). Παράλληλα, το Shogun υποστηρίζει τη χρήση προ-υπολογισμένων πυρήνων. Είναι επίσης δυνατό να χρησιμοποιηθεί ένας συνδυασμένος πυρήνας, δηλαδή ένας πυρήνας που αποτελείται από έναν γραμμικό συνδυασμό αυθαίρετων πυρήνων σε διαφορετικούς τομείς. Οι συντελεστές ή τα βάρη του γραμμικού συνδυασμού μπορούν επίσης να μαθευτούν από το αναπτυσσόμενο μοντέλο μηχανικής μάθησης. Για το σκοπό αυτό το Shogun προσφέρει μια λειτουργικότητα εκμάθησης πολλαπλών πυρήνων.

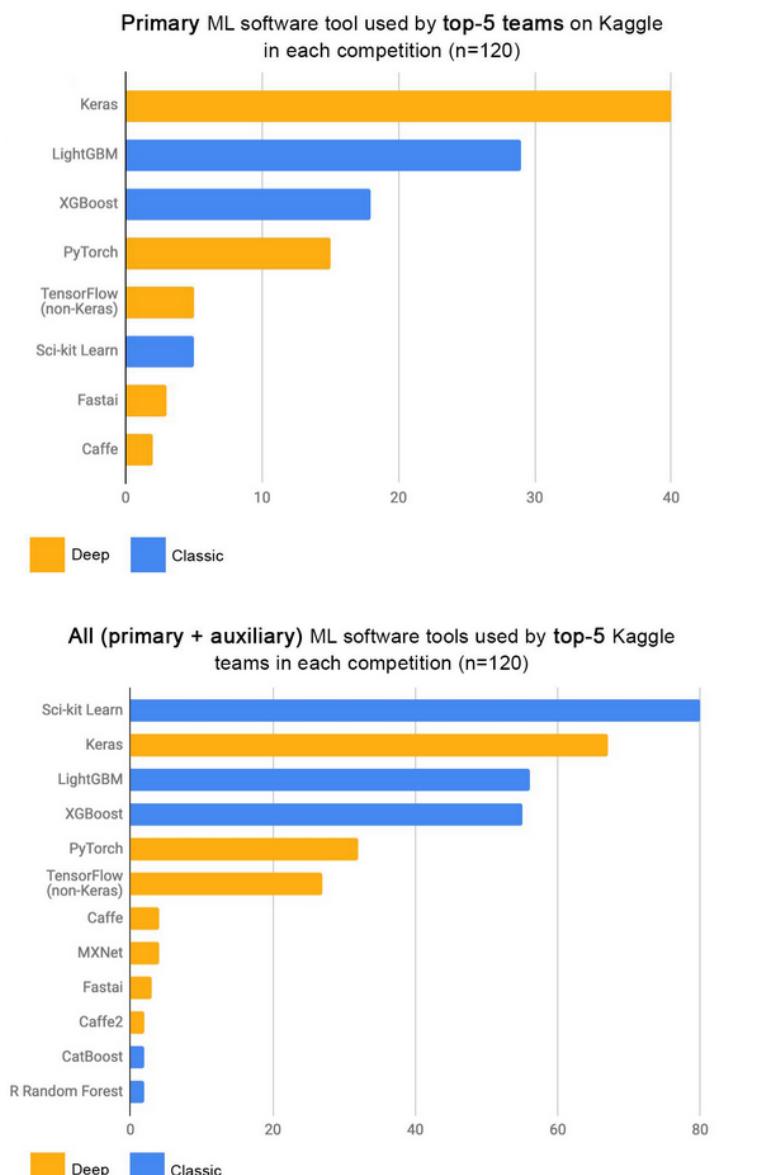
### 3.5.9. Keras.io

To Keras είναι μια βιβλιοθήκη νευρωνικών δικτύων ανοιχτού κώδικα γραμμένη σε Python, το οπίο αναπτύχθηκε στο πλαίσιο της ερευνητικής προσπάθειας του έργου ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System). Έχει τη δυνατότητα να τρέχει πάνω από άλλα ανεξάρτητα λογισμικά μηχανικής μάθησης όπως το TensorFlow, το Microsoft Cognitive Toolkit, το R, το Theano ή το PlaidML (Géron, 2019; keras.io., 2023; Vincent, Bengio, Chapados, & Delalleau., 2023). Είναι εύκολο στη χρήση, αρθρωτό και επεκτάσιμο δεδομένου ότι σχεδιάστηκε και αναπτύχθηκε έτσι ώστε να επιτρέπει γρήγορο πειραματισμό με βαθιά νευρωνικά δίκτυα (Géron, 2019; keras.io., 2023; Vincent et al., 2023).

To 2017, η ομάδα TensorFlow της Google αποφάσισε να υποστηρίξει το Keras και να το ενσωματώσει στην κύρια βιβλιοθήκη του λογισμικού μηχανικής μάθησης TensorFlow λόγω των αποδοτικών αποτελεσμάτων του (Géron, 2019; keras.io., 2023; Vincent et al., 2023). Ο ίδιος ο δημιουργός του Keras, ο François Chollet, εξήγησε ότι το Keras σχεδιάστηκε για να είναι μια διεπαφή και όχι ένα αυτόνομο λογισμικό μηχανικής μάθησης (Géron, 2019; keras.io., 2023; Vincent et al., 2023). Η Microsoft έχει προσθέσει επίσης ένα backend CNTK στο Keras, διαθέσιμο από την έκδοση CNTK 2.0 (keras.io, 2023; TensorFlow, 2023b).

Βασικά χαρακτηριστικά του λογισμικού αυτού είναι ότι μπορεί να χρησιμοποιηθεί για εύκολη και γρήγορη δημιουργία πρωτοτύπων μοντέλων μηχανικής μάθησης. Παράλληλα, το Keras υποστηρίζει δίκτυα συνέλιξης, βοηθά τα επαναλαμβανόμενα δίκτυα, ενώ μπορεί ακόμη να υποστηρίζει συνδυασμό δύο δικτύων. To Keras μπορεί να τρέξει σε CPU και GPU. Ανάμεσα στα πλεονεκτήματά του συμπεριλαμβάνονται ότι είναι φιλικό προς το χρήστη, Modular και Επεκτάσιμο. Τέλος, το εργαλείο αυτό είναι διαθέσιμο δωρεάν. Ωστόσο, ένα από τα βασικά μειονεκτήματά του είναι ότι για να μπορέσει ένας χρήστης να χρησιμοποιήσει το Keras ώστε να δημιουργήσει και να εκπαιδεύσει μοντέλα μηχανικής μάθησης, πρέπει οπωσδήποτε να κάνει χρήση άλλων λογισμικών όπως του TensorFlow, Theano ή CNTK.

Το Keras, ως λογισμικό, είναι ένα API σχεδιασμένο για ανθρώπους και όχι για μηχανές, δίνοντας προτεραιότητα στην εμπειρία του προγραμματιστή (user experience). Ταυτοχρόνως, ακολουθεί βέλτιστες πρακτικές για τη μείωση του γνωστικού φόρτου: ελαχιστοποιεί τον απαιτούμενο συνολικό αριθμό ενεργειών χρήστη, παρέχει συνεπή και απλά API, και τέλος προσφέρει σαφή και εφαρμόσιμη ανατροφοδότηση σε περίπτωση σφάλματος χρήστη. Αυτό κάνει το Keras εύκολο στην εκμάθηση και ακόμη πιο εύκολο και ευέλικτο στη χρήση. Αυτή η ευκολία χρήσης δεν συνεπάγεται το κόστος της μειωμένης ευελιξίας: επειδή το Keras ενσωματώνεται σε βάθος με τη λειτουργικότητα χαμηλού επιπέδου TensorFlow, δίνει τη δυνατότητα στο χρήστη να αναπτύξει ροές εργασίας υψηλής δυνατότητας hackable, όπου μπορεί να προσαρμοστεί οποιοδήποτε στοιχείο λειτουργικότητας. Το Keras κατατάχθηκε ως #1 για τη βαθιά μάθηση τόσο μεταξύ των πρωτογενών πλαισίων όσο και μεταξύ όλων των πλαισίων που χρησιμοποιήθηκαν:



Εικόνα 3-22: Αποτελέσματα έρευνας. (Πηγή: [https://keras.io/why\\_keras/](https://keras.io/why_keras/))

### 3.5.10. Σύγκριση λογισμικών Μηχανικής Μάθησης

**Πίνακας 2: Συγκριτικό διάγραμμα χαρακτηριστικών των λογισμικών που χρησιμοποιούνται ευρέως για μηχανική μάθηση.**

(<https://www.softwaretestinghelp.com/machine-learning-tools/>)

	<b>Πλατφόρμα</b>	<b>Κόστος</b>	<b>Γλώσσα προγραμματισμού</b>	<b>Αλγόριθμοι, χαρακτηριστικά</b>
<b>Scikit Learn</b>	Linux, Mac OS, Windows	Δωρεάν	Python, Cython, C, C++	Ταξινόμηση Οπισθοδρόμηση Ομαδοποίηση Προεπεξεργασία Επιλογή μοντέλου Μείωση διαστάσεων.
<b>PyTorch</b>	Linux, Mac OS, Windows	Δωρεάν	Python, C++, CUDA	Autograd Module Optim Module nn Module
<b>TensorFlow</b>	Linux, Mac OS, Windows	Δωρεάν	Python, C++, CUDA	Παροχή βιβλιοθήκης για dataflow programming.
<b>Weka</b>	Linux, Mac OS, Windows	Δωρεάν	Java	Προετοιμασία δεδομένων Ταξινόμηση Οπισθοδρόμηση Ομαδοποίηση Οραματισμός Κανόνες ένωσης εξόρυξης
<b>Colab</b>	Cloud Service	Δωρεάν	-	Υποστηρίζει βιβλιοθήκες για PyTorch, Keras, TensorFlow, and OpenCV
<b>Apache Mahout</b>	Cross-platform	Δωρεάν	Java Scala	Προ-επεξεργαστές Οπισθοδρόμηση Ομαδοποίηση Συστάτες Κατανεμημένη Γραμμική Άλγεβρα.
<b>Accors.Net</b>	Cross-platform	Δωρεάν	C#	Ταξινόμηση Οπισθοδρόμηση Κατανομή Ομαδοποίηση Δοκιμές υποθέσεων & Μέθοδοι πυρήνα Εικόνα, Ήχος
<b>Shogun</b>	Windows Linux UNIX Mac OS	Δωρεάν	C++	Οπισθοδρόμηση Ταξινόμηση Ομαδοποίηση Υποστήριξη διανυσματικά μηχανήματα.

---

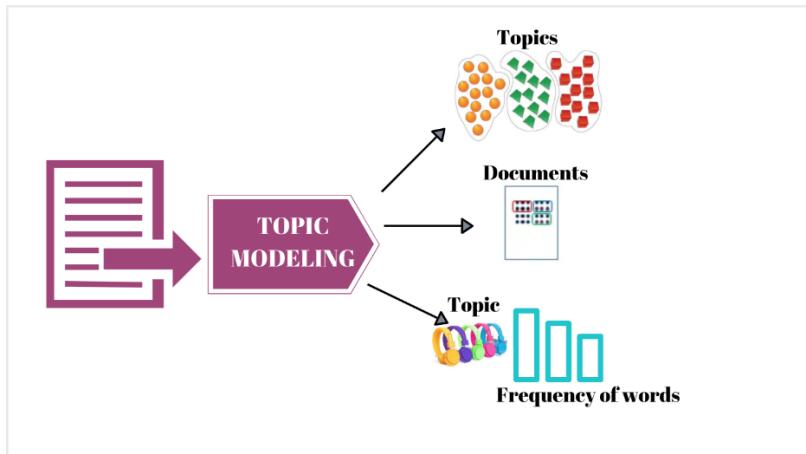
<b>Keras.io</b>	Cross-platform	Δωρεάν	Python	Μείωση διαστάσεων Διαδικτυακή μάθηση API για νευρωνικά δίκτυα
-----------------	----------------	--------	--------	---

---



## 4. Topic Modelling

Το Topic Modelling είναι μια τεχνική ανάλυσης κειμένου που χρησιμοποιείται στην επεξεργασία φυσικής γλώσσας (NLP) για τον εντοπισμό των θεμάτων ή των σημαντικών θεμάτων που εμφανίζονται σε ένα σύνολο κειμένων. Η βασική ιδέα είναι να ανακαλυφθούν κρυμμένες δομές θεμάτων μέσα σε ένα σύνολο κειμένων χωρίς την ανάγκη προκαθορισμένων θεμάτων ή κατηγοριών<sup>14</sup>.



Εικόνα 4-1: Παράδειγμα εφαρμογής Topic Modelling. (Πηγή: <https://medium.com/analytics-vidhya/how-to-perform-topic-modeling-using-mallet-abc43916560f>)

Η διαδικασία Topic Modelling αναλαμβάνει να εξάγει τα κύρια θέματα που ενδέχεται να είναι παρόντα σε ένα σύνολο κειμένων μέσω αλγορίθμων επεξεργασίας γλώσσας. Ο πιο γνωστός αλγόριθμος για Topic Modelling είναι το Latent Dirichlet Allocation (LDA)<sup>15</sup>. Ο LDA μοντελοποιεί κάθε κείμενο ως μία συλλογή θεμάτων που εκτιμάται ότι εμφανίζονται σε αυτό<sup>16</sup>. Κάθε θέμα αναπαρίσταται από μία κατανομή λέξεων.

Η διαδικασία του Topic Modelling μπορεί να παράσχει σημαντικές πληροφορίες για το περιεχόμενο των κειμένων χωρίς να απαιτείται προκαθορισμένη γνώση για τα θέματα που ενδέχεται να περιέχονται σε αυτά. Έτσι, μπορεί να χρησιμοποιηθεί για την ομαδοποίηση κειμένων, την ανίχνευση συναισθημάτων, την εξαγωγή στατιστικών πληροφοριών και άλλων εργασιών στην ανάλυση κειμένου.

Το Topic Modelling έχει εφαρμογές σε πολλούς τομείς, όπως η επεξεργασία και ανάλυση κειμένου στα μέσα κοινωνικής δικτύωσης, η ανάλυση αναφορών πελατών, η κατηγοριοποίηση ειδήσεων, η ανακάλυψη γνώσης από επιστημονικά άρθρα και η ανάλυση αποκλειστικών συνεντεύξεων.

Στα πλαίσια της παρούσας διπλωματικής εργασίας, η τεχνική Topic Modelling θα εφαρμοστεί με σκοπό την ανάπτυξη ενός συστήματος υποστήριξης αποφάσεων που βασίζεται στην ανάλυση συναισθήματος, παρέχοντας την κατάλληλη πληροφορία στους καταναλωτές για τη λήψη ενημερωμένων αποφάσεων αγοράς. Αυτό το σύστημα θα τους δώσει τη δυνατότητα να αντιμετωπίζουν τον όγκο των προϊόντων που είναι διαθέσιμα και να επιλέγουν αυτά που θα

<sup>14</sup> <https://medium.com/analytics-vidhya/how-to-perform-topic-modeling-using-mallet-abc43916560f>

<sup>15</sup> [https://en.wikipedia.org/wiki/Topic\\_model](https://en.wikipedia.org/wiki/Topic_model)

<sup>16</sup> [https://en.wikipedia.org/wiki/Latent\\_Dirichlet\\_allocation](https://en.wikipedia.org/wiki/Latent_Dirichlet_allocation)

ικανοποιήσουν καλύτερα τις ανάγκες και τις προτιμήσεις τους και θα προσφέρουν τη μεγαλύτερη ικανοποίηση.

#### 4.1. LDA topic modelling τεχνική

Ο LDA (Latent Dirichlet Allocation) είναι ένας αλγόριθμος μηχανικής μάθησης που χρησιμοποιείται για το *Topic Modelling*, μια τεχνική ανάλυσης κειμένου<sup>17</sup>. Ο στόχος του LDA είναι να ανακαλύψει τις κρυφές δομές θεμάτων που υπάρχουν σε ένα σύνολο κειμένων χωρίς προκαθορισμένες κατηγορίες ή θέματα.

Ας εξετάσουμε πώς λειτουργεί ο LDA<sup>18</sup>:

1. *Ορισμός αρχικών παραμέτρων:*

Καταρχήν, πρέπει να ορίσουμε τον αριθμό των θεμάτων που θέλουμε να εξάγουμε από το σύνολο κειμένων. Αυτός ο αριθμός μπορεί να είναι προκαθορισμένος ή να υπολογίζεται αυτόματα.

2. *Ανάθεση τυχαίων θεμάτων:*

Αρχικά, κάθε λέξη σε κάθε έγγραφο του συνόλου κειμένων αντιστοιχεί τυχαία σε ένα θέμα. Αυτός ο αντιστοιχισμός παρέχει μία αρχική εκτίμηση των θεμάτων που μπορεί να εμφανιστούν στα κείμενα.

3. *Εκτίμηση κατανομής λέξεων:*

Στη συνέχεια, ο αλγόριθμος υπολογίζει την πιθανότητα να εμφανίζεται μία συγκεκριμένη λέξη σε ένα θέμα και την πιθανότητα εμφάνισης ενός θέματος σε ένα έγγραφο. Αυτές οι κατανομές λέξεων είναι αρχικές εκτιμήσεις και αλλάζουν κατά τη διάρκεια της εκπαίδευσης του αλγορίθμου.

4. *Επανάληψη ανανέωσης θεμάτων:*

Σε αυτό το στάδιο, ο LDA επαναλαμβάνει τις παρακάτω διαδικασίες για μία συγκεκριμένη αριθμό επαναλήψεων:

- Υπολογίζει την πιθανότητα ένα θέμα να εμφανίζεται σε ένα έγγραφο.
- Υπολογίζει την πιθανότητα μία λέξη να ανήκει σε ένα θέμα.
- Επανατοποθετεί τις λέξεις σε νέα θέματα με βάση τις προηγούμενες εκτιμήσεις.

5. *Ανάκτηση τελικών θεμάτων:*

Μετά από αρκετές επαναλήψεις, ο αλγόριθμος συγκλίνει και επιστρέφει την τελική εκτίμηση των θεμάτων που ενδέχεται να εμφανίζονται στα κείμενα.

Ο LDA υποθέτει ότι τα κείμενα παράγονται από ένα μείγμα θεμάτων και ότι κάθε θέμα παράγει μία κατανομή λέξεων. Με αυτόν τον τρόπο, μπορεί να εντοπίσει τα κυριότερα θέματα που εμφανίζονται στα κείμενα και να παράσχει μία αναπαράσταση τους.

Ο LDA έχει ευρεία εφαρμογή στην ανάλυση κειμένου, συμπεριλαμβανομένων των ηλεκτρονικών μηνυμάτων, των κοινωνικών μέσων ενημέρωσης, των επιστημονικών άρθρων και της ανακάλυψης γνώσης από μεγάλα σύνολα δεδομένων.

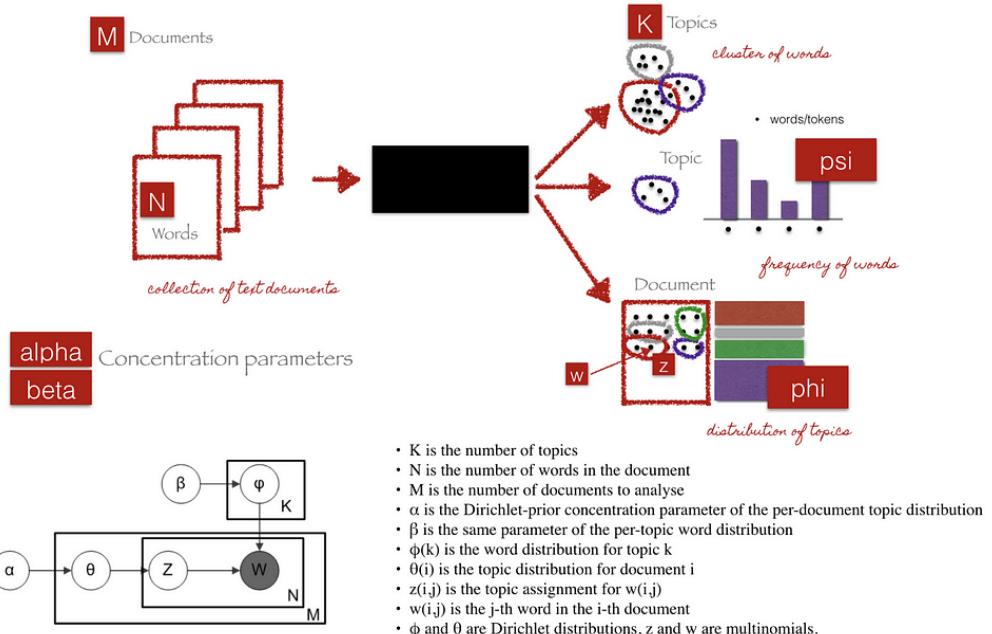
<sup>17</sup> [https://en.wikipedia.org/wiki/Latent\\_Dirichlet\\_allocation](https://en.wikipedia.org/wiki/Latent_Dirichlet_allocation)

<sup>18</sup> <https://towardsdatascience.com/latent-dirichlet-allocation-lda-9d1cd064ffa2>

#### 4.1.1. Βήμα προς βήμα παράδειγμα υλοποίησης του LDA (Latent Dirichlet Allocation) αλγορίθμου<sup>19</sup>

Ο LDA (Latent Dirichlet Allocation) αλγόριθμος είναι ένα παραγωγικό πιθανοτικό μοντέλο που υποθέτει ότι κάθε θέμα είναι ένα μείγμα πάνω από ένα υποκείμενο σύνολο λέξεων και κάθε έγγραφο είναι ένα μείγμα από πάνω από ένα σύνολο πιθανοτήτων θέματος<sup>20</sup>.

Μπορούμε να περιγράψουμε τη διαδικασία δημιουργίας του LDA, λαμβάνοντας υπόψη τον αριθμό  $M$  των εγγράφων, τον αριθμό  $N$  λέξεων και τον προηγούμενο αριθμό  $K$  θεμάτων (όπως απεικονίζεται στην Εικόνα 4-2), το μοντέλο εκπαιδεύεται στην έξοδο όπως ακολουθεί:



**Εικόνα 4-2: Παράδειγμα εφαρμογής LDA (Latent Dirichlet Allocation) αλγορίθμου. (Πηγή: <http://chdoig.github.io/pytexas2015-topic-modeling/#/3/4>)**

- ✓  $\psi$ , η κατανομή των λέξεων για κάθε θέμα  $K$
- ✓  $\phi$ , η κατανομή των θεμάτων για κάθε έγγραφο  $i$

#### Παράμετροι LDA

- (1) Η παράμετρος **άλφα** είναι η παράμετρος προηγούμενης συγκέντρωσης Dirichlet που αντιπροσωπεύει την πυκνότητα θέματος εγγράφου — με υψηλότερο άλφα, τα έγγραφα υποτίθεται ότι αποτελούνται από περισσότερα θέματα και έχουν ως αποτέλεσμα πιο συγκεκριμένη κατανομή θεμάτων ανά έγγραφο.
- (2) Η παράμετρος **βετα** είναι η ίδια παράμετρος προηγούμενης συγκέντρωσης που αντιπροσωπεύει την πυκνότητα θέματος-λέξεων — με υψηλή βήτα, τα θέματα θεωρείται ότι αποτελούνται από τις περισσότερες λέξεις και καταλήγουν σε μια πιο συγκεκριμένη κατανομή λέξεων ανά θέμα.

<sup>19</sup> <https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0>

<sup>20</sup> <https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0>

## Υλοποίηση LDA

Ο πλήρης κώδικας είναι διαθέσιμος ως Σημειωματάριο Jupyter στο GitHub<sup>21</sup>, και τα βήματα που ακολουθούμε για την υλοποίηση του αλγορίθμου είναι τα ακόλουθα:

1. Loading data / Φόρτωση δεδομένων
2. Data cleaning / Καθαρισμός δεδομένων
3. Exploratory analysis / Διερευνητική ανάλυση
4. Preparing data for LDA analysis / Προετοιμασία δεδομένων για ανάλυση LDA
5. LDA model training / Εκπαίδευση μοντέλου LDA
6. Analyzing LDA model results / Αναλύοντας τα αποτελέσματα του μοντέλου LDA

## Φόρτωση δεδομένων

Στη συγκεκριμένη εφαρμογή<sup>22</sup>, χρησιμοποιείται το σύνολο δεδομένων των άρθρων που δημοσιεύθηκαν στο συνέδριο NeurIPS (NIPS), το οποίο είναι ένα από τα πιο διάσημα ετήσια γεγονότα στην κοινότητα μηχανικής μάθησης. Το αρχείο δεδομένων CSV περιέχει πληροφορίες για τις διάφορες εργασίες NeurIPS που δημοσιεύθηκαν από το 1987 έως το 2016 (29 χρόνια!). Αυτές οι εργασίες συζητούν μια μεγάλη ποικιλία θεμάτων στη μηχανική μάθηση, από τα νευρωνικά δίκτυα έως τις μεθόδους βελτιστοποίησης και πολλά άλλα.

Αρχικά, το πρώτο βήμα είναι η μελέτη του περιεχομένου του αρχείου δεδομένων. Η Εικόνα 4-3 παρουσιάζει τα ακατέργαστα δεδομένα που χρησιμοποιήθηκαν για το συγκεκριμένο παράδειγμα υλοποίησης του LDA.

```
# Importing modules
import pandas as pd
import os

os.chdir('..')

# Read data into papers
papers = pd.read_csv('./data/NIPS Papers/papers.csv')

# Print head
papers.head()
```

Out[1]:	id	year	title	event_type	pdf_name	abstract	paper_text
	0	1 1987	Self-Organization of Associative Database and ...	NaN	1-self-organization-of-associative-database-and...	Abstract Missing	767\n\nSELF-ORGANIZATION OF ASSOCIATIVE DATABASE AND...
	1	10 1987	A Mean Field Theory of Layer IV of Visual Cortex	NaN	10-a-mean-field-theory-of-layer-iv-of-visual-cortex	Abstract Missing	683\n\nA MEAN FIELD THEORY OF LAYER IV OF VISUAL CORTTEX
	2	100 1988	Storing Covariance by the Associative Long-Term...	NaN	100-storing-covariance-by-the-associative-long-term...	Abstract Missing	394\n\nSTORING COVARIANCE BY THE ASSOCIATIVE LONG-TERM...
	3	1000 1994	Bayesian Query Construction for Neural Network...	NaN	1000-bayesian-query-construction-for-neural-network...	Abstract Missing	Bayesian Query Construction for Neural Network...
	4	1001 1994	Neural Network Ensembles, Cross Validation, an...	NaN	1001-neural-network-ensembles-cross-validation...	Abstract Missing	Neural Network Ensembles, Cross Validation, and...

**Εικόνα 4-3: Δείγμα ακατέργαστων δεδομένων. (Πηγή: <https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0>)**

<sup>21</sup>

<https://github.com/kapadias/medium-articles/blob/master/natural-language-processing/topic-modeling/Introduction%20to%20Topic%20Modeling.ipynb>

<sup>22</sup>

<https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0>

## Καθαρισμός Δεδομένων

Εφόσον ο στόχος αυτής της ανάλυσης είναι η εκτέλεση μοντελοποίησης θεμάτων, θα πρέπει να εστιάσουμε μόνο στα δεδομένα κειμένου από κάθε όρθρο και να αποθέσουμε άλλες στήλες μεταδεδομένων που δεν είναι χρήσιμα για τη συγκεκριμένη υλοποίηση.

```
# Remove the columns
papers = papers.drop(columns=['id', 'event_type', 'pdf_name'],
axis=1).sample(100)

# Print out the first rows of papers
papers.head()
```

	Out[2]:	year	title	abstract	paper_text
0	1987	Self-Organization of Associative Database and ...	Abstract Missing	767\n\nSELF-ORGANIZATION OF ASSOCIATIVE DATABASE...	
1	1987	A Mean Field Theory of Layer IV of Visual Cort...	Abstract Missing	683\n\nA MEAN FIELD THEORY OF LAYER IV OF VISU...	
2	1988	Storing Covariance by the Associative Long-Ter...	Abstract Missing	394\n\nSTORING COVARIANCE BY THE ASSOCIATIVE...	
3	1994	Bayesian Query Construction for Neural Network...	Abstract Missing	Bayesian Query Construction for Neural\nNetwor...	
4	1994	Neural Network Ensembles, Cross Validation, an...	Abstract Missing	Neural Network Ensembles, Cross\nValidation, a...	

*Αφαίρεση σημείων στίξης/κάτω περίβλημα*

Στη συνέχεια, οι δημιουργοί αυτής της υλοποίησης εκτελούν μια απλή προεπεξεργασία στο περιεχόμενο της στήλης paper\_text για να το κάνουν πιο επιδεκτικό για ανάλυση και αξιόπιστα αποτελέσματα. Για να το κάνουν αυτό, θα χρησιμοποιήσουν μια τυπική έκφραση για να αφαιρέσουν τυχόν σημεία στίξης και, στη συνέχεια, θα μετατρέψουν σε πεζά το κείμενο

```
# Load the regular expression library
import re

# Remove punctuation
papers['paper_text_processed'] = \
    papers['paper_text'].map(lambda x: re.sub('[,\.!?]', ' ', x))

# Convert the titles to lowercase
papers['paper_text_processed'] = \
    papers['paper_text_processed'].map(lambda x: x.lower())

# Print out the first rows of papers
papers['paper_text_processed'].head()
```

```
0    767\n\nself-organization of associative database...
1    683\n\nna mean field theory of layer iv of visual cortex...
2    394\n\nstoring covariance by the associative long-term storage...
3    bayesian query construction for neural\nnetworks...
4    neural network ensembles cross\\validation and...
```

Name: paper\_text\_processed, dtype: object

**Εικόνα 4-4: Παράδειγμα αφαίρεση σημείων στίξεως.**

## Διερευνητική Ανάλυση

Για να επαληθεύσουν εάν η προεπεξεργασία είναι αρκετή, θα δημιουργήσουν ένα σύννεφο λέξεων χρησιμοποιώντας το πακέτο wordcloud για να λάβουν μια οπτική αναπαράσταση των πιο κοινών λέξεων. Η διαδικασία αυτή είναι το κλειδί για την καλύτερη κατανόηση των δεδομένων, καθώς επίσης και για τη διασφάλιση ότι βρισκόμαστε στο σωστό δρόμο ή απαιτείται περαιτέρω προεπεξεργασία πριν από την εκπαίδευση του μοντέλου.

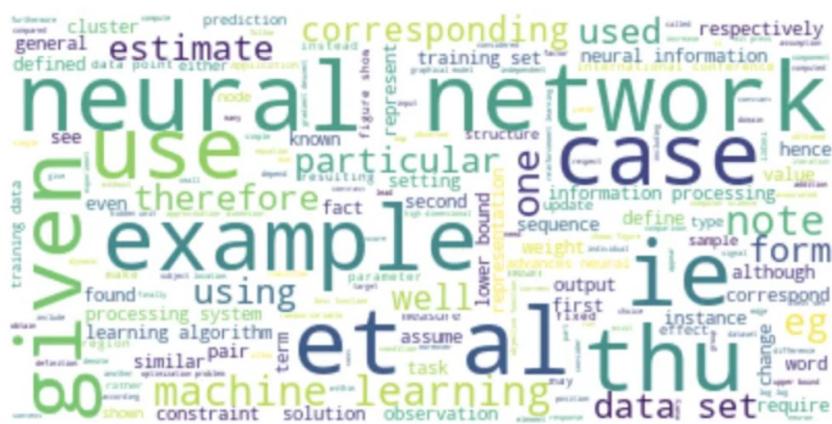
```
# Import the wordcloud library
from wordcloud import WordCloud

# Join the different processed titles together.
long_string = ','.join(list(papers['paper_text_processed'].values))

# Create a WordCloud object
wordcloud = WordCloud(background_color="white", max_words=5000,
contour_width=3, contour_color='steelblue')

# Generate a word cloud
wordcloud.generate(long_string)

# Visualize the word cloud
wordcloud.to_image()
```



## Προετοιμασία δεδομένων για ανάλυση LDA

Στη συνέχεια, οι δημιουργοί της υλοποίησης μετατρέπουν τα δεδομένα κειμένου σε μια μορφή που θα χρησιμεύσει ως είσοδος για το μοντέλο εκπαίδευσης LDA. Ξεκινάνε κάνοντας tokening το κείμενο και αφαιρώντας τα stopwords. Στη συνέχεια, μετατρέπουν το διακριτικό αντικείμενο σε σώμα και λεξικό, όπως απεικονίζεται στις Εικόνες που ακολουθούν.

```
import gensim
from gensim.utils import simple_preprocess
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords

stop_words = stopwords.words('english')
stop_words.extend(['from', 'subject', 're', 'edu', 'use'])

def sent_to_words(sentences):
    for sentence in sentences:
        # deacc=True removes punctuations
        yield(gensim.utils.simple_preprocess(str(sentence),
deacc=True))

def remove_stopwords(texts):
    return [[word for word in simple_preprocess(str(doc))
            if word not in stop_words] for doc in texts]

data = papers.paper_text_processed.values.tolist()
data_words = list(sent_to_words(data))

# remove stop words
data_words = remove_stopwords(data_words)

print(data_words[:1][0][:30])
```

Εικόνα 4-5: Παράδειγμα αφαίρεσης λέξεων τερματισμού.

```
[nltk_data] Downloading package stopwords to
[nltk_data]     /Users/shashank/nltk_data...
[nltk_data] Unzipping corpora/stopwords.zip.
['self', 'organization', 'associative', 'database', 'applications', 'hisashi', 'suzuki',
 'suguru', 'arimoto', 'osaka', 'university', 'toyonaka', 'osaka', 'japan', 'abstract',
 'efficient', 'method', 'self', 'organizing', 'associative', 'databases', 'proposed', 't
ogether', 'applications', 'robot', 'eyesight', 'systems', 'proposed', 'databases', 'ass
ociate']
```

```
import gensim.corpora as corpora
# Create Dictionary
id2word = corpora.Dictionary(data_words)
# Create Corpus
texts = data_words
# Term Document Frequency
corpus = [id2word.doc2bow(text) for text in texts]
# View
print(corpus[:1][0][:30])
```

```
[(0, 1), (1, 8), (2, 1), (3, 1), (4, 1), (5, 2), (6, 1), (7, 6), (8, 1), (9, 1), (10, 3
), (11, 1), (12, 2), (13, 2), (14, 1), (15, 1), (16, 1), (17, 1), (18, 1), (19, 6), (20
, 2), (21, 4), (22, 8), (23, 5), (24, 1), (25, 1), (26, 2), (27, 2), (28, 1), (29, 1)]
```

### **Εκπαίδευση μοντέλου LDA**

Οι δημιουργοί της υλοποίησης διατηρήσουν όλες τις παραμέτρους στις προεπιλογές, εκτός από την εισαγωγή του αριθμού των θεμάτων. Για τους σκοπούς αυτού του παραδείγματος, δημιουργούν ένα μοντέλο με 10 θέματα όπου κάθε θέμα είναι ένας συνδυασμός λέξεων-κλειδιών και κάθε λέξη-κλειδί συνεισφέρει με ένα συγκεκριμένο βάρος στο θέμα.

```
from pprint import pprint
# number of topics
num_topics = 10
# Build LDA model
lda_model = gensim.models.LdaMulticore(corpus=corpus,
                                         id2word=id2word,
                                         num_topics=num_topics)
# Print the Keyword in the 10 topics
pprint(lda_model.print_topics())
doc_lda = lda_model[corpus]
```

```

[(0,
  '0.008*"model" + 0.007*"data" + 0.005*"using" + 0.005*"set" + 0.005*"one" +
  '0.004*"algorithm" + 0.004*"learning" + 0.004*"models" + 0.004*"time" +
  '0.003*"distribution"),
(1,
  '0.006*"model" + 0.005*"learning" + 0.005*"two" + 0.004*"data" +
  '0.004*"time" + 0.004*"using" + 0.004*"algorithm" + 0.004*"figure" +
  '0.004*"set" + 0.004*"function"),
(2,
  '0.006*"algorithm" + 0.006*"model" + 0.005*"function" + 0.005*"network" +
  '0.005*"data" + 0.004*"using" + 0.004*"time" + 0.004*"learning" +
  '0.004*"one" + 0.003*"set"),
(3,
  '0.006*"learning" + 0.006*"function" + 0.005*"model" + 0.005*"data" +
  '0.005*"one" + 0.004*"algorithm" + 0.004*"set" + 0.004*"time" +
  '0.004*"using" + 0.003*"number"),
(4,
  '0.007*"model" + 0.007*"learning" + 0.006*"algorithm" + 0.005*"set" +
  '0.005*"data" + 0.004*"function" + 0.004*"using" + 0.004*"one" +
  '0.004*"figure" + 0.003*"time"),
(5,
  '0.008*"model" + 0.006*"algorithm" + 0.005*"data" + 0.005*"set" +
  '0.004*"function" + 0.004*"learning" + 0.004*"one" + 0.004*"used" +
  '0.003*"time" + 0.003*"also"),
(6,
  '0.005*"data" + 0.005*"function" + 0.004*"model" + 0.004*"algorithm" +
  '0.004*"learning" + 0.004*"using" + 0.004*"figure" + 0.004*"problem" +
  '0.003*"training" + 0.003*"two"),
(7,
  '0.009*"learning" + 0.007*"model" + 0.007*"data" + 0.005*"set" +
  '0.005*"network" + 0.004*"one" + 0.004*"algorithm" + 0.004*"number" +
  '0.004*"using" + 0.003*"log"),
(8,
  '0.009*"learning" + 0.006*"data" + 0.005*"algorithm" + 0.004*"function" +
  '0.004*"problem" + 0.004*"set" + 0.004*"using" + 0.004*"time" +
  '0.004*"two" + 0.004*"algorithm" + 0.004*"number" + 0.003*"problem" +
  '0.003*"function"]),
(9,
  '0.006*"model" + 0.005*"data" + 0.004*"learning" + 0.004*"one" +
  '0.004*"set" + 0.004*"algorithm" + 0.004*"number" + 0.003*"problem" +
  '0.003*"function")]

```

#### **Εικόνα 4-6: Εκπαίδευση μοντέλου LDA.**

### **Αναλύοντας τα αποτελέσματα του μοντέλου LDA**

Έχοντας εκπαιδεύσει το μοντέλο, το επόμενο βήμα είναι η οπτικοποίηση των θεμάτων με σκοπό την ερμηνευτικότητα του μοντέλου και των αποτελεσμάτων του. Για το σκοπό αυτό, οι δημιουργοί της συγκεκριμένης υλοποίησης χρησιμοποιούν ένα δημοφιλές πακέτο οπτικοποίησης, το pyLDAvis, το οποίο έχει σχεδιαστεί για να βοηθά διαδραστικά με:

- ✓ Καλύτερη κατανόηση και ερμηνεία μεμονωμένων θεμάτων και
- ✓ Καλύτερη κατανόηση των σχέσεων μεταξύ των θεμάτων.

Για το (1), μπορείτε να επιλέξετε χειροκίνητα κάθε θέμα για να δείτε τους πιο συχνούς και/ή «σχετικούς» όρους του, χρησιμοποιώντας διαφορετικές τιμές της παραμέτρου  $\lambda$ . Αυτό μπορεί να σας βοηθήσει όταν προσπαθείτε να εκχωρήσετε ένα ανθρώπινο ερμηνεύσιμο όνομα ή «νόημα» σε κάθε θέμα.

Για το (2), η εξερεύνηση του Διαθεματικού Σχεδίου Απόστασης μπορεί να σας βοηθήσει να μάθετε πώς σχετίζονται τα θέματα μεταξύ τους, συμπεριλαμβανομένης της πιθανής δομής υψηλότερου επιπέδου μεταξύ ομάδων θεμάτων.

```

import pyLDAvis.gensim
import pickle
import pyLDAvis

# Visualize the topics
pyLDAvis.enable_notebook()

LDAvis_data_filepath = os.path.join('./results',
/lda_vis_prepared_'+str(num_topics))

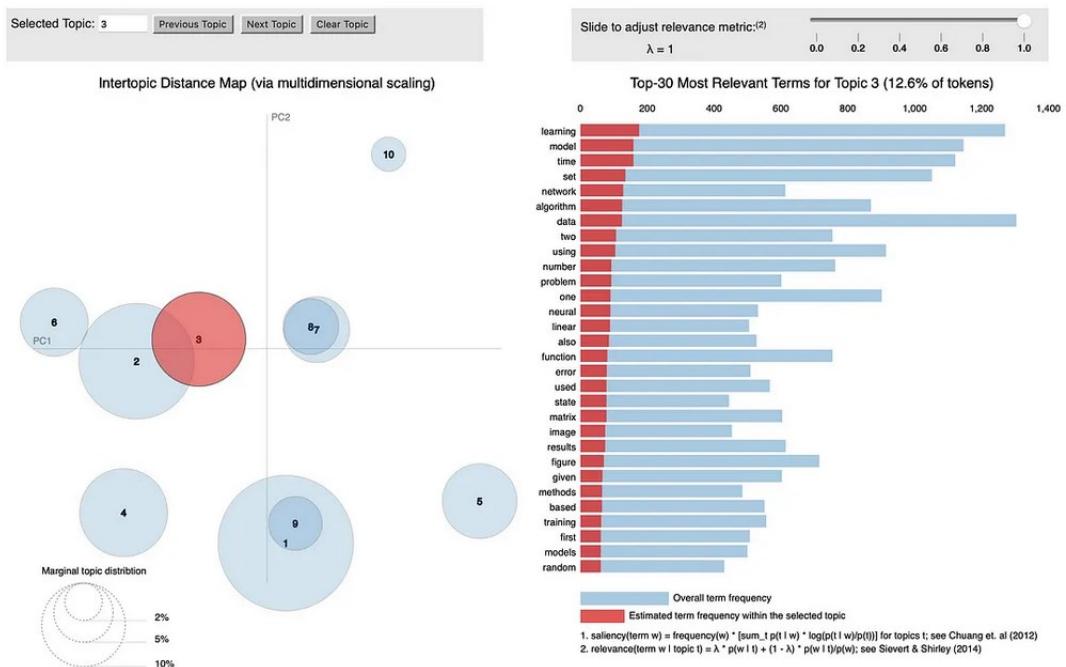
# # this is a bit time consuming - make the if statement True
# # if you want to execute visualization prep yourself
if 1 == 1:
    LDAvis_prepared = pyLDAvis.gensim.prepare(lda_model, corpus,
id2word)
    with open(LDAvis_data_filepath, 'wb') as f:
        pickle.dump(LDAvis_prepared, f)

# load the pre-prepared pyLDAvis data from disk
with open(LDAvis_data_filepath, 'rb') as f:
    LDAvis_prepared = pickle.load(f)

pyLDAvis.save_html(LDAvis_prepared, './results/lda_vis_prepared_'+
str(num_topics) +'.html')

LDAvis_prepared

```



Εικόνα 4-7: Οπτικοποίηση αποτελεσμάτων μοντέλου LDA.

Η μηχανική μάθηση έχει γίνει ολοένα και πιο δημοφιλής την τελευταία δεκαετία και οι πρόσφατες εξελίξεις στην υπολογιστική διαθεσιμότητα οδήγησαν σε εκθετική ανάπτυξη σε ανθρώπους που αναζητούν τρόπους με τους οποίους μπορούν να ενσωματωθούν νέες μέθοδοι για την προώθηση του τομέα της Επεξεργασίας Φυσικής Γλώσσας. Συχνά, αντιμετωπίζουμε τα μοντέλα θεμάτων ως αλγόριθμους μαύρου κουτιού. Ο στόχος αυτής της υποενότητας όπου εξηγεί βήμα προς βήμα ένα παράδειγμα υλοποίησης του αλγορίθμου LDA<sup>23</sup>, είναι να ρίξει φως στα υποκείμενα μαθηματικά και τις διαισθήσεις πίσω από αυτά, καθώς και τον κώδικα υψηλού

<sup>23</sup> <https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0>

επιπέδου. Στην επόμενη υποενότητα, θα εξετάσουμε μία άλλη τεχνική Topic Modelling ή οποία βασίζεται στο BERTopic.

#### 4.2. BERTopic topic modelling τεχνική

Η τεχνική BERTopic είναι μια προηγμένη μέθοδος Topic Modelling που χρησιμοποιεί το προπαρασκευασμένο μοντέλο BERT (Bidirectional Encoder Representations from Transformers) για την ανάλυση και ομαδοποίηση κειμένων. Ο BERT είναι ένα προηγμένο μοντέλο μάθησης βαθιάς μάθησης που έχει εκπαιδευτεί να κατανοεί και να αναπαράγει τη φυσική γλώσσα.

Ας εξετάσουμε πώς λειτουργεί η τεχνική BERTopic:

1. Εκπαίδευση του BERT μοντέλου:

Αρχικά, το μοντέλο BERT εκπαίδευται σε μεγάλα σύνολα δεδομένων, όπως τον περασμένο χρόνο. Αυτή η εκπαίδευση επιτρέπει στο μοντέλο να κατανοήσει την αναπαράσταση των λέξεων και της φράσης, καθώς και να αντιληφθεί το σημασιολογικό πλαίσιο και τις συνδέσεις μεταξύ των λέξεων.

2. Προεπεξεργασία των κειμένων:

Τα κείμενα που θα αναλυθούν υπόκεινται σε προεπεξεργασία προκειμένου να αφαιρεθούν άχρηστες λέξεις, να γίνει διαχωρισμός σε λέξεις και να εφαρμοστεί κανονικοποίηση, όπως αφαίρεση των κεφαλαίων γραμμάτων και αντικατάσταση των αριθμών με μια συγκεκριμένη αναπαράσταση, όπως το "#".

3. Ανάκτηση προεκπαιδευμένων εμβελειών (embeddings):

Στη συνέχεια, τα προεκπαιδευμένα εμβελέχθηκαν του μοντέλου BERT χρησιμοποιούνται για να αναπαραστήσουν τις λέξεις σε μια χωρική αναπαράσταση. Αυτός ο χώρος εμβέλειας διατηρεί τις σημασιολογικές συσχετίσεις μεταξύ των λέξεων.

4. Δημιουργία θεμάτων:

Στο στάδιο αυτό, χρησιμοποιείται ο αλγόριθμος Clustering για την ομαδοποίηση των κειμένων βάσει των εμβελειών. Ο αλγόριθμος εκχωρεί τα κείμενα σε θέματα με βάση την ομοιότητά τους, χρησιμοποιώντας την απόσταση μεταξύ των εμβελειών.

5. Εξαγωγή σημαντικών λέξεων:

Τέλος, για κάθε θέμα που προκύπτει, εξάγονται οι σημαντικές λέξεις που το αντιπροσωπεύουν. Αυτές οι λέξεις μπορούν να χρησιμοποιηθούν για να περιγράψουν το θέμα και να παρέχουν κατανόηση για το περιεχόμενο των κειμένων που ανήκουν σε αυτό το θέμα.

Η τεχνική BERTopic είναι ιδιαίτερα αποτελεσματική στην ανάλυση κειμένων, καθώς επωφελείται από την ισχύ του μοντέλου BERT στην κατανόηση της γλώσσας και την αναπαράσταση των λέξεων. Αυτό την καθιστά έναν ισχυρό αλγόριθμο για την αναγνώριση και ομαδοποίηση θεμάτων σε μεγάλα σύνολα δεδομένων κειμένων.

Υπάρχουν πολλά οφέλη από τη χρήση του BERTopic για μοντελοποίηση θεμάτων:

- Βελτιωμένη ποιότητα θέματος: Το BERTopic έχει αποδειχθεί ότι παράγει πιο συνεκτικά και ερμηνεύσιμα θέματα σε σύγκριση με τις παραδοσιακές τεχνικές μοντελοποίησης θεμάτων, όπως η λανθάνουσα κατανομή Dirichlet (LDA).

- Καλύτερος χειρισμός μεγάλων συλλογών κειμένου: Το BERTTopics μπορεί να χειριστεί αποτελεσματικά μεγάλες συλλογές κειμένου, κάτι που είναι ζωτικής σημασίας για τα σύγχρονα δεδομένα κειμένου.
- Δυνατότητα καταγραφής σημασιολογικών σχέσεων μεταξύ λέξεων: Το BERTTopics αξιοποιεί τη δύναμη αναπαράστασης των μετασχηματιστών για να καταγράψει τις σημασιολογικές σχέσεις μεταξύ των λέξεων στο κείμενο, γεγονός που οδηγεί σε πιο ακριβή και ουσιαστικά θέματα.
- Λεπτός έλεγχος του αριθμού των θεμάτων: Σε αντίθεση με άλλες τεχνικές μοντελοποίησης θεμάτων, το BERTTopics παρέχει λεπτομερή έλεγχο του αριθμού των θεμάτων που θα εξαχθούν, κάτι που μπορεί να είναι χρήσιμο σε εφαρμογές όπου ο αριθμός των θεμάτων είναι κρίσιμος.
- Εύκολος συντονισμός: Το BERTTopics εκπαιδεύεται σε μεγάλα σώματα κειμένου, πράγμα που σημαίνει ότι μπορεί να βελτιωθεί σε συγκεκριμένες συλλογές κειμένου, οδηγώντας σε βελτιωμένη απόδοση σε αυτές τις συλλογές.

#### 4.2.1. Βήμα προς βήμα παράδειγμα υλοποίησης του BERTopic αλγορίθμου<sup>24</sup>

Το παρακάτω παράδειγμα είναι ένα άρθρο που εξηγεί τη χρήση του BERTopic, μιας τεχνικής για το topic modeling στον τομέα της επεξεργασίας φυσικής γλώσσας (NLP). Ο συγγραφέας εξηγεί την έννοια του BERTopic και τους λόγους για τους οποίους αξίζει να χρησιμοποιηθεί για το topic modeling. Έπειτα, παρουσιάζει ένα παράδειγμα εφαρμογής του BERTopic σε ένα σύνολο δεδομένων από κριτικές εστιατορίων στο Kaggle, όπου κατηγοριοποιούνται οι κριτικές σε διάφορες σημαντικές κατηγορίες όπως: «ατμόσφαιρα», «φαγητό», «προσωπικό», «εξυπηρέτηση» και ούτω καθεξής.

Ο συγγραφέας παρουσιάζει την επεξεργασία των δεδομένων, τον υπολογισμό των embeddings των κειμένων και την εκπαίδευση του μοντέλου BERTopic. Στη συνέχεια, αναλύει τα θέματα που παράχθηκαν και τους υπολογισμούς που μπορούν να γίνουν για την κατανόησή τους. Τέλος, αναφέρει πως μπορούμε να μειώσουμε τις εκτός κλάσης (outliers) εγγραφές και παρουσιάζει μεθόδους οπτικοποίησης των θεμάτων και των κριτικών. Συγκεκριμένα, ακολουθεί τα παρακάτω βήματα:

##### 1 Προετοιμασία των δεδομένων:

Αρχικά, πρέπει να προετοιμάσουμε τα δεδομένα μας. Αυτό περιλαμβάνει την καθαρισμό και την προεπεξεργασία των κειμένων. Μπορούμε να αφαιρέσουμε περίτεχνους χαρακτήρες, σημεία στίξης, συνδυασμούς αριθμών και λέξεων κ.λπ. Επίσης, μπορούμε να εφαρμόσουμε την τοκεντοποίηση (tokenization) για να μετατρέψουμε τα κείμενα σε λίστα από λέξεις.

##### 2 Υπολογισμός των embeddings:

Στο επόμενο βήμα, πρέπει να υπολογίσουμε τα embeddings των κειμένων μας. Χρησιμοποιούμε το BERT (Bidirectional Encoder Representations from Transformers) για να αναπαραστήσουμε τα κείμενα σε μια πολυδιάστατη χώρα. Το BERT είναι ένα προ-εκπαιδευμένο μοντέλο που έχει εκπαιδευτεί σε μεγάλο όγκο

---

<sup>24</sup> <https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8>

δεδομένων της γλώσσας. Οι embeddings αντιπροσωπεύουν τη σημασία και τη δομή των κειμένων.

3 Εκπαίδευση του μοντέλου BERTopic:

Στο επόμενο βήμα, εκπαιδεύουμε το μοντέλο BERTopic με τα δεδομένα μας. Αυτό συμπεριλαμβάνει την εκτέλεση του αλγορίθμου BERTopic πάνω στα embeddings που υπολογίσαμε προηγουμένως. Ο αλγόριθμος χρησιμοποιεί τη μέθοδο συσταδοποίησης για να αναγνωρίσει τα θέματα στα δεδομένα.

4 Ανάκτηση των θεμάτων:

Μετά την εκπαίδευση, μπορούμε να ανακτήσουμε τα θέματα που αναγνωρίστηκαν από το μοντέλο. Κάθε θέμα αναπαρίσταται από έναν κεντρικό όρο και μια λίστα από συναφείς όρους. Μπορούμε επίσης να λάβουμε τον βαθμό συσχέτισης (coherence score) για κάθε θέμα, ο οποίος μας δίνει μια ένδειξη του πόσο συνεκτικό είναι το θέμα.

5 Οπτικοποίηση των θεμάτων:

Τέλος, μπορούμε να οπτικοποιήσουμε τα θέματα που αναγνωρίστηκαν. Αυτό μπορεί να γίνει με τη χρήση γραφημάτων ή άλλων μεθόδων οπτικοποίησης για να εμφανίσουμε τη συσχέτιση μεταξύ των θεμάτων ή τη συγχόνητα εμφάνισή τους στα δεδομένα.

Αυτά είναι τα βασικά βήματα που περιλαμβάνονται στο παράδειγμα χρήσης του BERTopic. Με αυτόν τον τρόπο, μπορούμε να αναγνωρίσουμε και να οργανώσουμε θέματα σε ένα σύνολο δεδομένων κειμένων με βάση το BERTopic. Στη συνέχεια, τα βήματα αυτά επεξηγούνται λεπτομερώς.

#### Ρύθμιση σημειωματάριου και εισαγωγή των δεδομένων

Πριν ξεκινήσουμε, εισάγουμε τα πακέτα που χρειαζόμαστε.

```
from bertopic import BERTopic
from sentence_transformers import SentenceTransformer, util
from umap import UMAP
```

```
import os
import pandas as pd

df = pd.read_csv('Restaurant_Reviews.tsv', sep='\t')
```

**Εικόνα 4-8: Ρύθμιση σημειωματάριου και εισαγωγή των δεδομένων.**

#### Εισαγωγή και εξερεύνηση της φύσης του συνόλου δεδομένων

Στην εικόνα που ακολουθεί, παρουσιάζονται μερικά δείγματα από το σύνολο δεδομένων.

	Review object	Liked int64 0 - 1	
	I love this pla... 0.2% I won't be ba... 0.2% 994 others 99.6%		
0	Wow... Loved this place.	1	
1	Crust is not good.	0	
2	Not tasty and the texture was just...	0	
3	Stopped by during the late May bank...	1	
4	The selection on the menu was...	1	
5	Now I am getting angry and I want...	0	

**Εικόνα 4-9: Εισαγωγή και εξερεύνηση της φύσης του συνόλου δεδομένων.**

Πρόκειται για πολύ καλά ισορροπημένα δεδομένα, πιθανώς σκοπίμως δειγματοληπτικά. Τα δεδομένα αποτελούνται από 500 εγγραφές αρνητικών σχολίων (με τιμή 0) και 500 εγγραφές θετικών σχολίων (με τιμή 1) .

df.Liked.value_counts()	
0	500
1	500
Name: Liked, dtype: int64	
Hosted on  Deepnote	

**Εικόνα 4-10: Οπτικοποίηση δεδομένων.**

Τα ισορροπημένα δεδομένα αναφέρονται σε μια κατάσταση όπου ο αριθμός των παραδειγμάτων σε κάθε κατηγορία ή κλάση μιας μηχανής μάθησης είναι περίπου ισοδύναμος ή παρόμοιος. Σε άλλα λόγια, κάθε κατηγορία έχει παρόμοιο αριθμό παραδειγμάτων που την αντιπροσωπεύουν.

Η έλλειψη ισορροπίας στα δεδομένα μπορεί να έχει αρνητικές επιπτώσεις στην απόδοση του μοντέλου μηχανικής μάθησης. Όταν οι κατηγορίες έχουν ανισόροπο αριθμό παραδειγμάτων, το μοντέλο τείνει να μάθει καλύτερα την κατηγορία με το μεγαλύτερο αριθμό παραδειγμάτων και να αγνοεί ή να κατατάσσει εσφαλμένα τις κατηγορίες με το μικρότερο αριθμό παραδειγμάτων. Αυτό μπορεί να οδηγήσει σε ανεπιθύμητη συμπεριφορά του μοντέλου κατά τη διάρκεια της πρόβλεψης.

Για να αντιμετωπιστεί η έλλειψη ισορροπίας, μπορούν να εφαρμοστούν διάφορες τεχνικές όπως η υπερδειγματοληψία (oversampling) και η υποδειγματοληψία (undersampling). Η υπερδειγματοληψία περιλαμβάνει τη δημιουργία νέων παραδειγμάτων από τις κατηγορίες με το μικρότερο αριθμό παραδειγμάτων, ενώ η υποδειγματοληψία αφαιρεί ή μειώνει τον αριθμό παραδειγμάτων από τις κατηγορίες με το μεγαλύτερο αριθμό παραδειγμάτων. Ο στόχος είναι να επιτευχθεί μια ισορροπημένη συλλογή δεδομένων, έτσι ώστε το μοντέλο να μπορεί να εκπαιδευτεί και να γενικεύσει σωστά για όλες τις κατηγορίες.

### Έλεγγος της έκτασης των εγγράφων (ή των κειμένων)

Στη συνέχεια, υπολογίζουμε πόσοι χαρακτήρες υπάρχουν σε κάθε εγγραφή. Ο μέγιστος αριθμός χαρακτήρων είναι 149. Συνεπώς, είναι δυνατή η αποτελεσματική χρήση του Sentence Transformer ως μοντέλου για την ενσωμάτωση, καθώς υπάρχουν ως επί το πλείστον σύντομες παράγραφοι. Πριν προχωρήσουμε στο επόμενο βήμα, ας μετατρέψουμε την "Επισκόπηση" σε λίστα (μπορείτε επίσης να αποθηκεύσετε τα έγγραφα ως numpy array).

```
df['len_chac'] = df.Review.str.len()
df.len_chac.describe()
```

```
count    1000.000000
mean      58.315000
std       32.360052
min       11.000000
25%      33.000000
50%      51.000000
75%      80.000000
max      149.000000
Name: len_chac, dtype: float64
```

```
docs = df.Review.to_list()
```

### Προεπεξεργασία/προετοιμασία δεδομένων

Αρχικά, πρέπει να προετοιμάσουμε τα δεδομένα μας. Αυτό περιλαμβάνει την καθαρισμό και την προεπεξεργασία των κειμένων. Μπορούμε να αφαιρέσουμε περίτεχνους χαρακτήρες, σημεία στίξης, συνδυασμούς αριθμών και λέξεων κ.λπ. Επίσης, μπορούμε να εφαρμόσουμε την τοκεντοποίηση (tokenization) για να μετατρέψουμε τα κείμενα σε λίστα από λέξεις.

Σε αντίθεση με πολλές παραδοσιακές μεθόδους NLP όπου πρέπει να αφαιρέσουμε λέξεις τερματισμού (ένα σύνολο λέξεων που χρησιμοποιούνται συνήθως σε οποιαδήποτε γλώσσα). Για παράδειγμα, στα αγγλικά, "the", "is" και "and"), η τεκμηρίωση του BERTopic προτείνει να μην αφαιρεθούν οι λέξεις αυτές. Αντιθέτως, οι λέξεις τερματισμού πριν από τα έγγραφα χρησιμοποιούνται για τη δημιουργία των ενσωματώσεων σύμφωνα με την τεχνική BERTopic.

Οι λέξεις τερματισμού (stop words) αναφέρονται σε κοινές λέξεις που συνήθως έχουν χαμηλή πληροφοριακή αξία και εμφανίζονται συχνά σε ένα κείμενο. Αυτές οι λέξεις περιλαμβάνουν συνδέσμους (όπως "και", "ή", "αλλά"), αρθρα (όπως "ο", "η", "το"), αντωνυμίες (όπως "εγώ", "εσύ", "αυτός") και άλλες λέξεις κοινής χρήσης.

Η αφαίρεση των λέξεων τερματισμού από ένα κείμενο μπορεί να γίνει για διάφορους λόγους:

1. Μείωση του μεγέθους του κειμένου: Η αφαίρεση των λέξεων τερματισμού μειώνει τον αριθμό των λέξεων στο κείμενο, οδηγώντας σε μικρότερο όγκο δεδομένων και πιο αποδοτική επεξεργασία.
2. Βελτίωση της απόδοσης των αλγορίθμων επεξεργασίας κειμένου: Ορισμένοι αλγόριθμοι επεξεργασίας κειμένου, όπως οι αλγόριθμοι ανάλυσης συναισθημάτων ή ανάλυσης θεμάτων, επηρεάζονται αρνητικά από τις λέξεις τερματισμού. Αφαιρώντας αυτές τις λέξεις, μπορεί να βελτιωθεί η ακρίβεια και η αξιοπιστία των αλγορίθμων.

Ωστόσο, αξίζει να σημειωθεί ότι η αφαίρεση των λέξεων τερματισμού δεν είναι πάντα απαραίτητη ή επιθυμητή σε όλες τις περιπτώσεις επεξεργασίας κειμένου. Σε ορισμένα πεδία, όπως η ανάλυση συναισθημάτων, μπορεί να είναι χρήσιμο να διατηρηθούν ορισμένες λέξεις τερματισμού που μπορούν να περιλαμβάνουν χρήσιμη πληροφορία για την ανίχνευση συναισθημάτων.

Στη συγκεκριμένη περίπτωση, η κατάργηση των λέξεων διακοπής ως βήμα προεπεξεργασίας δεν συνιστάται, καθώς τα μοντέλα ενσωμάτωσης που βασίζονται σε μετασχηματιστή που χρησιμοποιούμε χρειάζονται το πλήρες πλαίσιο προκειμένου να δημιουργήσουν ακριβείς ενσωματώσεις. Αντίθετα, μπορούμε να χρησιμοποιήσουμε το CountVectorizer για την προεπεξεργασία των εγγράφων μας αφού δημιουργήσουμε ενσωματώσεις και ομαδοποιήσουμε τα έγγραφά μας. Δεν υπάρχουν σχεδόν μειονεκτήματα στη χρήση του CountVectorizer για την αφαίρεση λέξεων τερματισμού. Με αυτόν τον τρόπο στο BERTopic οι ενσωματώσεις θε δημιουργούνται με βάση τα πλήρη κείμενα.

```
from sklearn.feature_extraction.text import CountVectorizer
vectorizer_model = CountVectorizer(stop_words="english")
```

Hosted on  Deepnote

**Εικόνα 4-11: Προεπεξεργασία/προετοιμασία δεδομένων.**

Στη συνέχεια, υπολογίζουμε εκ των προτέρων τις ενσωματώσεις, έτσι ώστε οι ενσωματώσεις να μπορούν να επαναχρησιμοποιηθούν χωρίς να απαιτείται ο επανυπολογισμός τους. Αυτό είναι βολικό, ειδικά εάν τα έγγραφά που χρησιμοποιούμε είναι τεράστια, γεγονός που οδηγεί σε μακρύ υπολογισμό των ενσωματώσεων:

- **model\_embedding**: μοντέλο ενσωμάτωσης που βασίζεται στο Sentence Transformer
- **corpus\_embeddings**: ενσωματώσεις που δημιουργούνται από τα έγγραφα (δηλαδή κριτικές από τους πελάτες)

Ακολουθούν επεξηγήσεις για ορισμένες από τις παραμέτρους που χρησιμοποιούνται για αυτό το μοντέλο:

- Η παράμετρος **n\_gram\_range** αναφέρεται στο CountVectorizer που χρησιμοποιείται κατά τη δημιουργία της αναπαράστασης θέματος. Σχετίζεται με τον αριθμό των λέξεων που θέλετε στην αναπαράσταση του θέματός σας. Για παράδειγμα: το γρήγορο φαγητό είναι ένα n-γραμμάριο 2.
- Η παράμετρος **nr\_topics** καθορίζει, μετά την εκπαίδευση του μοντέλου θέματος, τον αριθμό των θεμάτων που θα μειωθούν. Για παράδειγμα, εάν το μοντέλο θέματός έχει ως αποτέλεσμα 50 θέματα αλλά έχουμε ορίσει το nr\_topics σε 10, τότε το μοντέλο θέματος θα προσπαθήσει να μειώσει τον αριθμό των θεμάτων με βάση τον καθορισμένο ακέραιο. Σε αυτήν την περίπτωση, χρησιμοποιούμε το "auto" για αυτόματη μείωση των θεμάτων χρησιμοποιώντας το HDBSCAN.
- Η παράμετρος **min\_topic\_size** αποτελεί σημαντική παράμετρος, η οποία χρησιμοποιείται για να καθορίσει ποιο μπορεί να είναι το ελάχιστο μέγεθος ενός θέματος. Όσο μικρότερη είναι αυτή η τιμή τόσο περισσότερα θέματα δημιουργούνται. Εάν ορίσουμε αυτήν την τιμή πολύ υψηλή, τότε είναι πιθανό να μην δημιουργηθούν απλά θέματα! Για το λόγο αυτό θα πρέπει να ρυθμίσουμε αυτήν την τιμή πολύ χαμηλή

ώστε να λάβουμε πολλές μικροσυστάδες (microclusters). Το να βρούμε το κατάλληλο min\_topic\_size αποτελεί ζήτημα δοκιμής και εξέταση σφάλματος. Θα πρέπει πρώτα που θα πρέπει πρώτα να δημιουργήσω το αρχικό μοντέλο για να το ελέγξω και να επιστρέψω εδώ αργότερα να δοκιμάσω με διαφορετικό.

```
%%time
model = BERTopic(
    n_gram_range=(1, 2),
    vectorizer_model=vectorizer_model,
    nr_topics='auto',
    min_topic_size=10,
    diversity=0.7,
    seed_topic_list=[
        ["experience", "bad", "good", "nice"],
        ["place", "atmosphere", "toilet", "clean"],
        ["staff", "waitress", "service"],
        ["wait", "time", "long"],
        ["food", "taste"]
    ],
    calculate_probabilities=True).fit(docs, corpus_embeddings)
```

CPU times: user 20.3 s, sys: 164 ms, total: 20.4 s  
Wall time: 20.6 s

Hosted on  Deepnote

**Εικόνα 4-12: Προεπεξεργασία/προετοιμασία δεδομένων.**

Μετά την εκπαίδευση του μοντέλου, μπορούμε να δημιουργήσουμε το προβλεπόμενο θέμα και τις πιθανότητες (για το πόσο σίγουρο είναι το θέμα που έχει εκχωρηθεί) για κάθε ένα από τα έγγραφα.

```
topics, probabilities = model.transform(docs, corpus_embeddings)
```

### Κατανόηση των θεμάτων που δημιουργούνται

Μέχρι αυτό το σημείο, δημιουργήσαμε τα διάφορα θέματα για κάθε αρχείο (κριτική εστιατορίου) με βάση τε δεδομένα μας. Ωστόσο, δεδομένου ότι το μοντέλο μας είναι μοντέλο μάθησης χωρίς επίβλεψη (unsupervised learning model), είναι απαραίτητο οι ίδιοι οι χρήστες του μοντέλου να το ερμηνεύσουν και να κατανοήσουν τα θέματα (δηλαδή τις συστάσεις/clusters) που δημιουργήθηκαν. Η τεχνική BERTopic προσφέρει μία χρήσιμη συνάρτηση (get\_topic\_freq) για τη δημιουργία της συχνότητας εμφάνισης των θεμάτων. Για το παράδειγμά μας, η συνάρτηση δίνει τον πίνακα που ακολουθεί. Παρατηρώντας τον πίνακα, καταλήγουμε στις παρακάτω παρατηρήσεις:

- Το θέμα -1 (Topic -1 στην εικόνα) αποτελεί τους outliers. Αυτό σημαίνει ότι αυτές οι 329 κριτικές (~33% των 1000) δεν μπορούν να ταξινομηθούν σε καμία συστάδα.

- Συνολικά δημιουργήθηκαν 9 θέματα. Για την οπτικοποίηση όλων των θεμάτων που δημιουργήθηκαν, ένας εύκολος τρόπος είναι να τρέξουμε την ακόλουθη εντολή `len(df_topic_freq) - 1`.



**Εικόνα 4-13: Συχνότητα εμφάνισης θεμάτων.** (Πηγή: <https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8>)

Έχοντας εκπαιδεύσει το BERTopic μοντέλο μας με βάση τα κείμενα (1000 κριτικές), μπορούμε να μελετήσουμε τα παραγόμενα θέματα ώστε να κατανοήσουμε γιατί παράχθηκαν τα συγκεκριμένα από το μοντέλο μας. Ωστόσο, κάτι τέτοιο χρειάζεται αρκετό χρόνο, ενώ παράλληλα δεν προσφέρει μία ολική αναπαράσταση των παραγόμενων θεμάτων. Η τεχνική BERTopic προσφέρει ένα εύκολο τρόπο οπτικοποίησης των παραγόμενων θεμάτων, ο οποίος είναι αρκετά παρόμοιος με την τεχνική LDAvis<sup>25</sup>

```
model.visualize_topics()
```

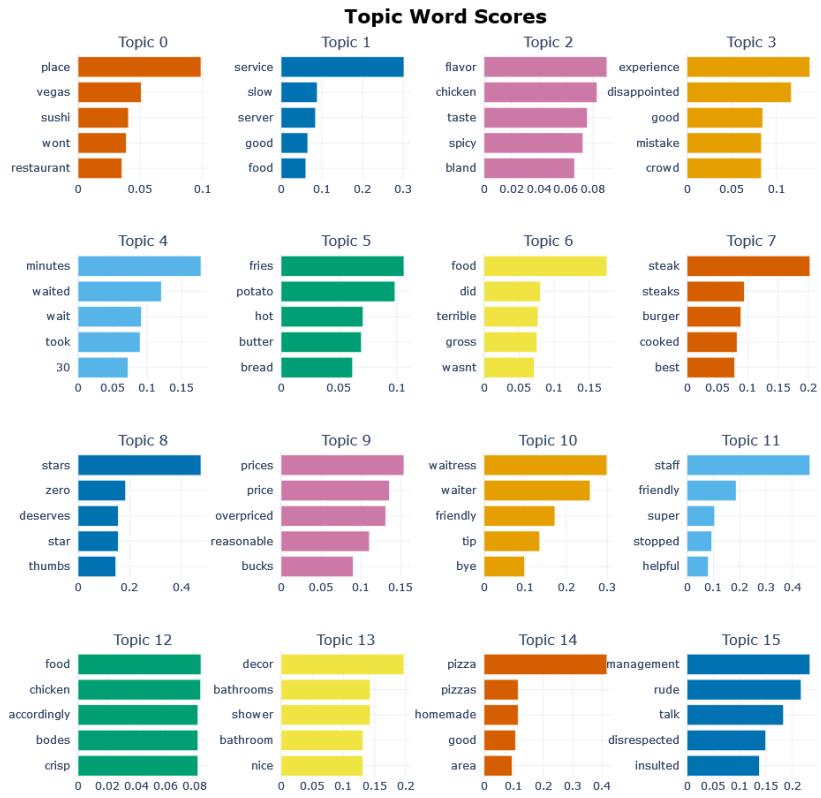
<sup>25</sup> <https://github.com/cpsievert/LDAvis>



**Εικόνα 4-14: Οπτικοποίηση των μοντέλου. (Πηγή: <https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8> )**

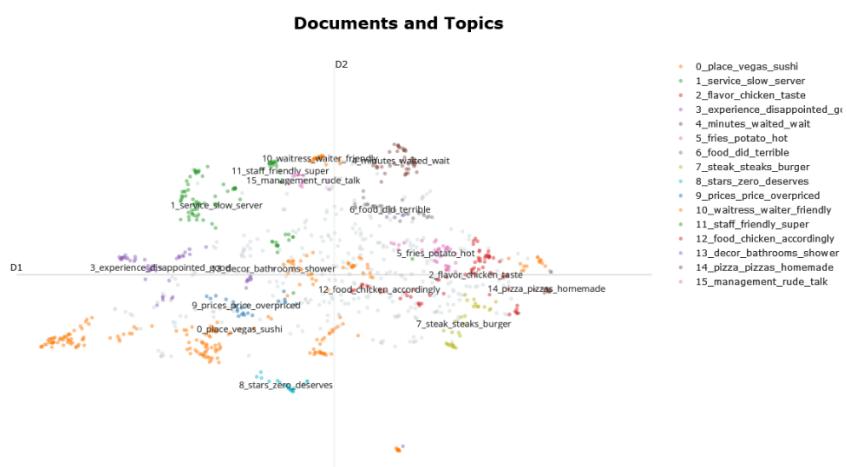
Ένας διαφορετικός τρόπος οπτικοποίησης των αποτελεσμάτων είναι μέσω διαγραμμάτων bar charts για κάθε αναπαράσταση θέματος. Τα διαγράμματα αυτά θα μας δείξουν τις N πιο αντιπροσωπευτικές λέξεις για κάθε διαφορετικό θέμα. Για παράδειγμα, αναφερόμενοι στο Θέμα 1 με λέξεις κλειδιά : service , slow , server, είναι λογικό να αναμένουμε ότι οι κριτικές σε αυτό το θέμα είναι σχετικές με την εξυπηρέτηση (“service-related”). Έτσι, μπορούμε να κατανοήσουμε καλύτερα και πιο αποτελεσματικά το κάθε θέμα. Αυτό είναι αρκετά χρήσιμο για την επαλήθευση των αποτελεσμάτων. Για τη συγκεκριμένη οπτικοποίηση, θα πρέπει να τρέξουμε την παρακάτω εντολή:

```
model.visualize_barchart(top_n_topics=topics_count)
```



**Εικόνα 4-15: Topic Bar Chart with Topic Word Scores. (Πηγή: <https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8> )**

Σε περίπτωση που χρειαζόμαστε πιο αναλυτικά αποτελέσματα, μπορούμε να προχωρήσουμε με μία πιο αναλυτική μέθοδο. Χρησιμοποιώντας τη συνάρτηση `topic_model.visualize_documents()`, μπορούμε να επανυπολογίσουμε τις ενσωματώσεις κειμένου (document embeddings), μειώνοντας τες σε έναν χώρο 2 διαστάσεων για πιο εύκολη οπτικοποίηση από το χρήστη.

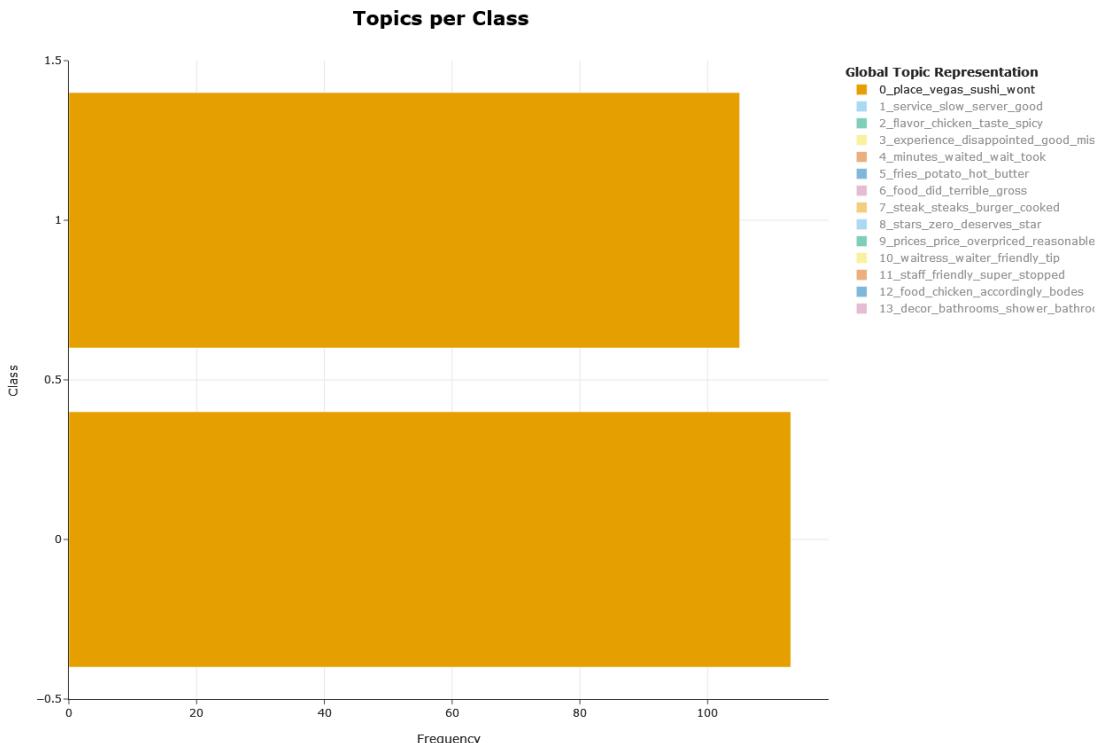


**Εικόνα 4-16: Ενσωματώσεις κειμένου σε 2 διαστάσεις. (Πηγή: <https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8> )**

## Οπτικοποίηση θεμάτων ανά κλάση

Στην περίπτωση που θέλουμε να οπτικοποιήσουμε τα παραγόμενα θέματα ανά κλάση, για το συγκεκριμένο παράδειγμα με τις κριτικές εστιατορίων, έχουμε δύο διαφορετικές κλάσεις: '0' για αρνητική κρητική, και '1' για θετική κρητική. Η συγκεκριμένη οπτικοποίηση αυτής της κατηγοριοποίησης (0/1) μας επιτρέπει, για κάθε διαφορετικό θέμα, να μπορούμε να αναγνωρίζουμε εάν το κάθε αρχείο (κρητική) αποτελεί αρνητική ή θετική κρητική. Με άλλα λόγια, η κατηγοριοποίηση αυτή αποτελεί την οπτικοποίηση των παραγόμενων θεμάτων κάτω από τις δύο διαφορετικές κλάσεις (0/1).

```
# visualize the topic representation of major topics per class:  
topics_per_class = model.topics_per_class(docs, classes=df.Liked.to_list())  
model.visualize_topics_per_class(topics_per_class, top_n_topics=14)
```



## Μείωση Outliers

Στα στατιστικά, οι outliers (ακραίες τιμές) αναφέρονται σε παρατηρήσεις ή τιμές που αποκλίνουν σημαντικά από τον μέσο όρο της υπόλοιπης δείγματος. Οι outliers μπορεί να είναι είτε εξαιρετικά μεγάλες είτε εξαιρετικά μικρές τιμές σε σχέση με τις υπόλοιπες παρατηρήσεις.

Οι outliers μπορεί να προκληθούν από διάφορες αιτίες, όπως ακραίες συμπεριφορές, ασυνήθιστες συνθήκες ή ακόμα και σφάλματα μέτρησης. Είναι σημαντικό να ανιχνεύονται και να διερευνώνται οι outliers, καθώς μπορεί να έχουν σημαντική επίδραση στα στατιστικά αποτελέσματα και την ερμηνεία των δεδομένων.

Οι outliers μπορούν να ανιχνευθούν με τη χρήση διάφορων μεθόδων. Μία κοινή προσέγγιση είναι η χρήση του κριτήριου των εκτός κλίμακας τριών τυπικών αποκλίσεων. Σύμφωνα με αυτό το κριτήριο, μια τιμή θεωρείται outlier εάν βρίσκεται εκτός τριών τυπικών αποκλίσεων από τον μέσο όρο.

Όταν ανιχνεύονται outliers, μπορούν να ληφθούν διάφορα μέτρα. Αυτά περιλαμβάνουν την αφαίρεση των outliers από το σύνολο δεδομένων, την αντικατάσταση τους με άλλες τιμές (όπως το μέσο όρο) ή την ανάλυση των outliers ξεχωριστά για περαιτέρω μελέτη.

Συνολικά, η ανίχνευση και διαχείριση των outliers αποτελεί σημαντικό βήμα στην ανάλυση δεδομένων και τη στατιστική, καθώς επιτρέπει την αποτύπωση πιο ακριβών και αντιπροσωπευτικών αποτελεσμάτων.

Παρατηρώντας τον πίνακα με τις συχνότητες θεμάτων, θα δούμε πως σχεδόν πάντα υπάρχουν outlier documents τα οποία δεν μπορούν να κατηγοριοποιηθούν κάτω από κανένα από τα παραγόμενα θέματα. Αυτά τα αρχεία έχουν κατηγοριοποιηθεί ως -1. Στο συγκεκριμένο παράδειγμα, όπως είδαμε και παραπάνω, τα outlier documents αποτελούν περίπου το 30% των συνολικών κριτικών.

```
# Comment out this line below if you decided to use the
#"probabilities" strategy
new_topics = model.reduce_outliers(docs, topics, strategy="c-tf-idf")

# Reduce outliers using the `probabilities` strategy (Uncomment to
use this)
#new_topics = model.reduce_outliers(docs, topics,
probabilities=probabilities, strategy="probabilities")
```

Εξαρτάται πάντα από την εφαρμογή, αλλά γενικότερα υπάρχουν δύο επιλογές σχετικά με το χειρισμό των outlier documents, οι οποίες αναλύονται παρακάτω:

1. Η πρώτη επιλογή είναι να προσπαθήσουμε να μειώσουμε τον αριθμό των αρχείων που έχουν κατηγοριοποιηθεί ως outliers. Η τεχνική BERTopic προσφέρει διάφορες συναρτήσεις για τη μείωση των outliers<sup>26</sup>. Η πιο διαδομένη συνάρτηση είναι η reduce\_outliers. Για την αποτελεσματική εκτέλεσή της, αρκεί να περάσουμε τα αρχεία μας και τα παραγόμενα θέματα. Στην εικόνα που ακολουθεί, μπορούμε να παρατηρήσουμε πως με την εκτέλεση της συνάρτησης reduce\_outliers, δεν υπάρχουν πλέον outliers στα δεδομένα μας (αρχεία κατηγοριοποιημένα ως -1) όταν παράγουμε τον πίνακα με τη συχνότητα των παραγόμενων θεμάτων/
2. Η δεύτερη επιλογή είναι να αγνοήσουμε ή να διαγράψουμε εντελώς τα αρχεία που έχουν κατηγοριοποιηθεί ως outliers. Στην περίπτωση αυτή, δεν προβλέπεται κάποια επιπλέον πράξη. Αυτό συμβαίνει, καθώς δύνανται μετά τη μείωση των outliers (1<sup>η</sup> επιλογή), να συνεχίσουμε να έχουμε έναν υψηλό αριθμό outliers στα αρχεία μας ή να παρατηρήσουμε ότι τα outliers αρχεία κατηγοριοποιήθηκαν με χαμηλή ακρίβεια. Οι outliers συνήθως αποτελούν ακραίες τιμές και πολλές φορές βέλτιστη λύση είναι η εκπαίδευση του μοντέλου χρησιμοποιώντας πιο αξιόπιστα δεδομένα, τα οποία έχουν κατηγοριοποιηθεί σωστά από το μοντέλο σε ένα από τα παραγόμενα θέματα.

<sup>26</sup> <https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8>

```
# This line is to update the model with the latest topic assignment  
model.update_topics(docs, topics=new_topics, vectorizer_model=vectorizer_model)
```

model.get_topic_freq()			
	Topic int64 -1 - 15	Count int64 11 - 275	
0	0	275	
1	1	101	
2	2	87	
3	3	77	
4	6	66	
5	4	53	
6	5	53	
7	7	52	
8	9	46	
9	12	32	

Hosted on  Deepnote

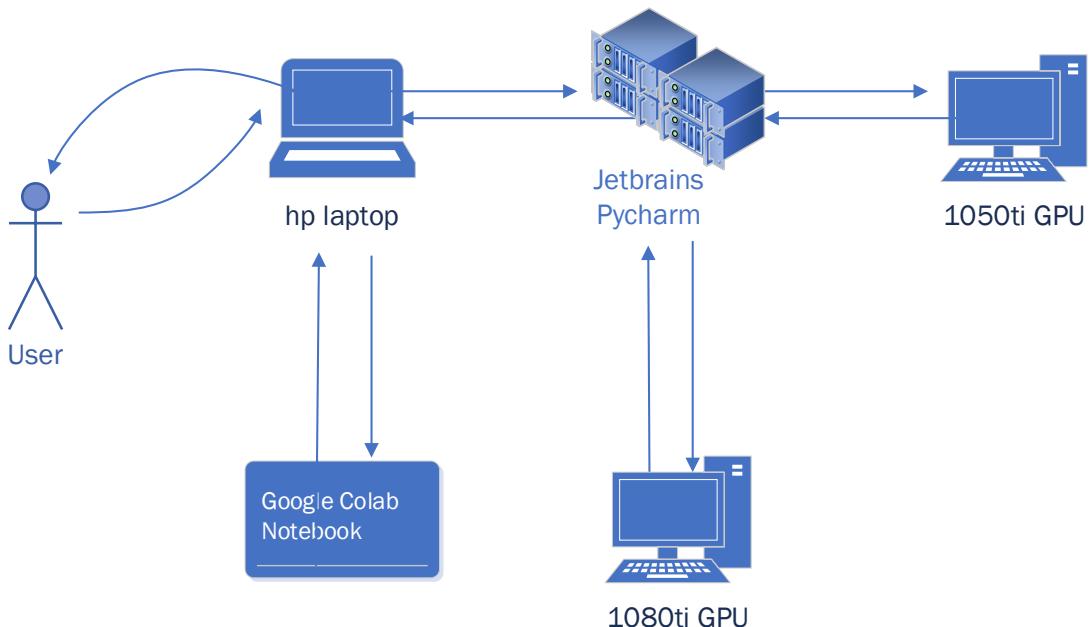
**Εικόνα 4-17:** Συχνότητα εμφάνισης θεμάτων μετά τη μείωση των outliers. (Πηγή:  
<https://medium.com/@nick-tan/topic-modeling-with-bertopic-a-cookbook-with-an-end-to-end-example-part-1-3ef739b8d9f8> )

## 5. Τεχνική Περιγραφή Υλοποίησης

Στο παρόν κεφάλαιο, θα περιγράψουμε την τεχνική υλοποίηση του συστήματος υποστήριξης αποφάσεων για την αγορά προϊόντων που μελετήθηκε και αναπτύχθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας, με έμφαση στην αρχιτεκτονική του συστήματος και την υλοποίηση του. Κατά την υλοποίηση, αξιοποιήθηκαν αλγόριθμοι μηχανικής μάθησης με στόχο τη δημιουργία ενός αποδοτικού και αποτελεσματικού συστήματος που θα παρέχει υποστήριξη στη λήψη αποφάσεων βασισμένη σε ανάλυση συναισθήματος.

Μέσω αυτού του συστήματος, προσφέρεται υποστήριξη στη λήψη αποφάσεων για την αγορά προϊόντων με βάση την ανάλυση των συναισθηματικών εκφράσεων των χρηστών. Η υλοποίηση αυτή αποτελεί ένα βήμα προς την αυτοματοποίηση και βελτίωση της διαδικασίας λήψης αποφάσεων στον τομέα των αγορών προϊόντων.

### 5.1. Αρχιτεκτονική συστήματος



**Εικόνα 5-1: Αρχιτεκτονική συστήματος.**

Το σύστημα από αναπτύχθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας για την υποστήριξη αποφάσεων για την αγορά προϊόντων βασισμένο σε ανάλυση συναισθήματος, αποτελείται από τα παρακάτω στοιχεία, όπως απεικονίζονται και στην Εικόνα που προηγείται:

- **HP Laptop:**
  - AMD Ryzen 3 2200U with Radeon Vega Mobile Gfx 2.50GHz
  - Ram 8.0GB
- **DESKTOP 1050Ti:**
  - AMD Ryzen 5 1600X Six-Core Processor 3.60GHz
  - RAM 16.0 GB
  - NVIDIA GeForce GTX 1050Ti
- **DESKTOP 1080Ti:**
  - AMD Ryzen 5 3600X Six-Core Processor 3.60GHz
  - RAM 16.0 GB
  - NVIDIA GeForce GTX 1080Ti
- **JetBrains Pycharm, JetBrains Toolbox, SSH Connection**

- **Google Colab Notebook and GPU**

Η κύρια υλοποίηση έγινε στο hp laptop καθώς υπήρχε σύνδεση μεταξύ των 2 desktop μέσω του JetBrains Toolbox το οποίο χρησιμοποιούσε την υπολογιστική ισχύ των desktop.

## 5.2. Web Scraper

Αρχικά για να συλλεχθούν τα δεδομένα και να δημιουργηθεί το dataset κατασκευάστηκε ένας web scraper.

Για την δημιουργία του web scraper χρησιμοποιήθηκε η βιβλιοθήκη beautifulsoup και webdriver της selenium για το headless firefox. Το headless firefox ήταν αναγκαίο καθώς από κανονικό browser μετά από κάποια requests η Amazon σταματούσε την σύνδεση, και δεν επέτρεπε την αποστολή επιπλέον δεδομένων, ενώ παράλληλα τα δεδομένα της Amazon φορτώνονται δυναμικά μέσω Javascript και όχι html.

Δίνοντας ως αρχικό όρισμα στην εφαρμογή την landing page της εφαρμογής (πρώτη σελίδα ακουστικών στην Amazon), ο web scraper βρίσκει και αποθηκεύει σε αρχείο κάθε url επόμενης σελίδας με πολλαπλά προϊόντα η κάθε μια εντοπίζοντας το href του next page κουμπιού στο κάτω μέρος της σελίδας. Κατά την διάρκεια της αναζήτησης των href διαγράφονται duplicates συνδέσμων που μπορεί να οδηγήσουν σε κάποιο dead end.

```
links = []
for url in products:
    print('~~~~~')
    print(url)

    options = Options()
    options.add_argument('--disable-blink-features=AutomationControlled')
    options.headless = True
    driver = webdriver.Firefox(firefox_binary=r"/usr/bin/firefox", options=options)

    driver.get(url)
    html_doc = driver.page_source
    driver.quit()
    soup = BeautifulSoup(html_doc, 'html.parser')

    for i in soup.find_all("a", {"class": "a-link-emphasis a-text-bold"}):
        print(i.get('href'))
        links.append(i.get('href'))

links = list(set(links)) # removes duplicates

urls = []
for i in links:
    i = 'https://www.amazon.com' + i
    # j += 1
    urls.append(i)

with open(r"urls.txt", 'a') as fp:
    fp.write('\n'.join(urls) + '\n')
```

**Εικόνα 5-2: Snippet κώδικα για τη δημιουργία του web scraper.**

Έχοντας κάθε σελίδα με τα προϊόντα ο web scraper βρίσκει για κάθε ένα από τα προϊόντα το αντίστοιχο href του που οδηγεί στην κάθε μία τους σελίδα. Κάθε url αποθηκεύεται σε αρχείο και διαγράφονται duplicates παραδείγματος χάριν προτεινόμενων προϊόντων ώστε να μην

παρθούν πολλαπλές φορές. Για κάθε url αρχικά αποθηκεύεται η τελική και μοναδική για κάθε ένα κατάληξη του href και μετά προστίθεται το πρόθεμα της σελίδας Amazon.

Μπαίνοντας στην κάθε σελίδα κάθε αποθηκευμένου url προϊόντος εντοπίζεται το μέρος της Javascript το οποίο εμφανίζει τα reviews των χρηστών. Κάθε review αποθηκεύεται σε αρχείο μορφής λίστας. Σε κάθε νέο προϊόν που προστίθεται αποθηκεύεται πριν τα reviews το url του προϊόντος σαν αναγνωριστικό και για μελλοντική χρήση.

Επιπλέον αποθηκεύεται η βαθμολογία κάθε review, η οποία υπάρχει σε μορφή αστερίσκων. Για την αποθήκευση της ο web scraper εντοπίζει πάλι το μέρος της Javascript που υπάρχουν οι αστερίσκοι από το ένα έως το πέντε και τα αποθηκεύει σε αντίστοιχο αρχείο.

### 5.3. Datasets - Data Preparation

Έχοντας σε αρχεία τα reviews και τα ratings κάθε διαφορετικής κρητικής καταναλωτή δημιουργούνται dataframes για την περαιτέρω και καλύτερη αξιοποίηση των δεδομένων που συλλέχθηκαν. Για την δημιουργία των dataframes χρησιμοποιείται η βιβλιοθήκη pandas. Η συγκεκριμένη βιβλιοθήκη θεωρείται η μακράν καλύτερη για την διαχείριση δεδομένων στην Python.

Διαγράφοντας κενά μεταξύ γραμμών που υπάρχουν από την αποθήκευση μέσω web scraper και διατηρώντας την σειρά καθώς και τα urls των προϊόντων δημιουργούνται 2 καθαρά dataframes, ένα με τα reviews και ένα με την αξιολόγηση αντιστοιχισμένη στην ίδια θέση με το review της.

Τέλος δημιουργείται ένα ξεχωριστό dataframe με 3 στήλες. Η πρώτη στήλη περιέχει το url του προϊόντος, η δεύτερη και Τρίτη περιλαμβάνουν τις κριτικές και τις αντίστοιχες αξιολογήσεις. Ο λόγος που δημιουργήθηκε αυτό το dataframe είναι για μελλοντική χρήση στην περίπτωση που κάνοντας αναζήτηση για κάποιο συγκεκριμένο προϊόν, να μπορεί να γίνει αποτελεσματική ταξινόμηση αποτελεσμάτων και εμφάνιση τους.

### 5.4. Topic Modelling

#### 5.4.1. LDA

Η τεχνική topic modelling LDA αναλύθηκε στο Κεφάλαιο 4.1 μαζί με ένα βήμα-προς-βήμα παράδειγμα. Στο συγκεκριμένο υποκεφάλαιο, θα περιγράψουμε πως αυτή η τεχνική χρησιμοποιήθηκε για τους σκοπούς διεκπεραίωσης της παρούσας διπλωματικής εργασίας.

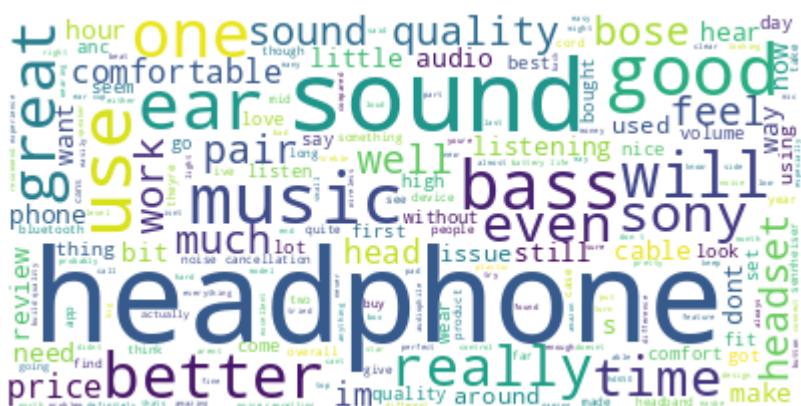
Περνώντας στο θέμα του Topic Modelling, παρακάτω υλοποιήθηκε αλγόριθμος Latent Dirichlet Allocation (LDA). Για την υλοποίηση του χρησιμοποιήθηκε η βιβλιοθήκη της SkLearn και συγκεκριμένα ο decomposition αλγόριθμος LatentDirichletAllocation της. Πριν οποιαδήποτε χρήση της παραπάνω βιβλιοθήκης ήταν αναγκαία η προεργασία του dataset. Χρησιμοποιήθηκαν τα reviews από το προαναφερθέν dataset και στην επεξεργάστηκαν.

Αρχικά διαγράφηκαν ειδικοί χαρακτήρες όπως ‘@’, ‘/’ καθώς και τα stopwords με την βιβλιοθήκη stopwords της nltk, όπως ‘the’, ‘is’, ‘and’ και τα κείμενα έγιναν όλα σε lowercase format. Αυτή είναι η πλέον καθορισμένη προεργασία για χρήση Latent Dirichlet Allocation και δεν γινόταν να υλοποιηθεί δίχως αυτή.

	review	stars	processed
0	They are great for the price and fit perfectly...	4.0	they are great for the price and fit perfectly...
1	Wife loves these for plugging into Roku remote...	4.0	wife loves these for plugging into roku remote...
2	Excellent quality, sound, and yet, they are no...	4.0	excellent quality, sound, and yet, they are no...
3	A very short review: These are wonderful headp...	4.0	a very short review: these are wonderful headp...
4	Buenos!!!	4.0	buenos!!!

**Εικόνα 5-3: Απεικόνιση προ-επεξεργασίας δεδομένων.**

Έχοντας έτοιμα τα ορίσματα και χρησιμοποιώντας την βιβλιοθήκη Wordcloud οπτικοποιούμε την συγχόνηση και σημαντικότητα των λέξεων με την διαφοροποίηση σε μέγεθος και χρώμα.



Εικόνα 5-4: Οπτικοποίηση σημαντικότερων λέξεων στις κριτικές (δεδομένα προς επεξεργασία).

Με μια πρώτη ματιά ξεχωρίζουν λέξεις όπως headphone, sound, music και bass, λέξεις άκρως αναμενόμενες σε προιόντα ακουστικών. Αναμένουμε να τις δούμε στην υλοποίηση του αλγόριθμου Latent Dirichlet Allocation.

Συνεχίζοντας με την προετοιμασία των reviews για την υλοποίηση του αλγόριθμου Latent Dirichlet Allocation, δημιουργούμε tokens για κάθε λέξη. Το tokenization είναι τελικής μορφής unicode strings τα οποία δεν θα δεχτούν περαιτέρω επεξεργασία και για την μορφοποίηση τους χρησιμοποιείται μέθοδος simple preprocess της gensim.

Tα reviews έχουν έρθει στην παρακάτω μορφή και είναι έτοιμα για το επόμενο βήμα του vectorization:

```
['great', 'price', 'fit', 'perfectly', 'guy', 'small', 'head']
```

Με την χρήση των μεθόδων corpora της gensim βιβλιοθήκης δημιουργούνται dictionaries με τα reviews σε μορφή διανυσμάτων. Σε κάθε διάνυσμα υπάρχει ένα mapping της κάθε λέξης σε όλο το λεξικό καθώς και η συχνότητα εμφάνισης του σε αυτό. Το κάθε διάνυσμα φαίνεται όπως το παρακάτω:

```
[ (0, 1), (1, 1), (2, 1), (3, 1), (4, 1), (5, 1), (6, 1) ]
```

Τέλος, έχοντας έτοιμα τα dictionaries και κάνοντας χρήση του μοντέλου Latent Dirichlet Allocation εμφανίζονται 5 κυρίαρχα topics με λέξεις οι οποίες είχαν εμφανιστεί και στο Wordcloud.

```
[ (0,
  '0.019*"sound" + 0.017*"headphones" + 0.007*"like" + 0.007*"music" + '
  '0.006*"noise" + 0.006*"bass" + 0.006*"really" + 0.005*"also" + '
  '0.005*"would" + 0.005*"im"),
(1,
  '0.015*"headphones" + 0.011*"sound" + 0.008*"like" + 0.007*"quality" + '
  '0.007*"good" + 0.006*"great" + 0.006*"ear" + 0.005*"noise" + 0.005*"get" + '
  '0.005*"bass"),
(2,
  '0.020*"headphones" + 0.012*"sound" + 0.009*"bass" + 0.007*"like" + '
  '0.006*"music" + 0.006*"quality" + 0.006*"good" + 0.005*"really" + '
  '0.005*"headline" + 0.005*"better"),
(3,
  '0.026*"headphones" + 0.018*"sound" + 0.010*"good" + 0.009*"quality" + '
  '0.008*"like" + 0.007*"noise" + 0.006*"bass" + 0.006*"music" + '
  '0.006*"better" + 0.005*"ear"),
(4,
  '0.015*"headphones" + 0.015*"sound" + 0.009*"good" + 0.008*"quality" + '
  '0.007*"like" + 0.006*"noise" + 0.005*"ear" + 0.005*"would" + 0.005*"better" +
  '0.005*"well")]

```

**Εικόνα 5-5: Οπτικοποίηση των 5 κυρίαρχων topics σύμφωνα με τον αλγόριθμο LDA.**

#### 5.4.2. BERTopic

Όπως έχει προαναφερθεί η τεχνική BERTopic είναι μια προηγμένη μέθοδος Topic Modelling που χρησιμοποιεί το προπαρασκευασμένο μοντέλο BERT (Bidirectional Encoder Representations from Transformers) για την ανάλυση και ομαδοποίηση κειμένων.

Γενικά στο BERTopic δεν χρειάζεται η προ-επεξεργασία των data αλλά καθώς το dataset που χρησιμοποιήθηκε είναι αρκετά απλό και γύρω από συγκεκριμένο θέμα, τα stopwords δημιουργούσαν πρόβλημα. Για αυτό τον λόγο διαγράφηκαν και έγινε lemmatization στο dataset.

Για την υλοποίηση του BERTopic χρησιμοποιείται η βιβλιοθήκη BERTopic και η UMAP. Δίνοντας σαν ορίσματα τα lemmatized reviews εμφανίζονται τα παρακάτω topic.

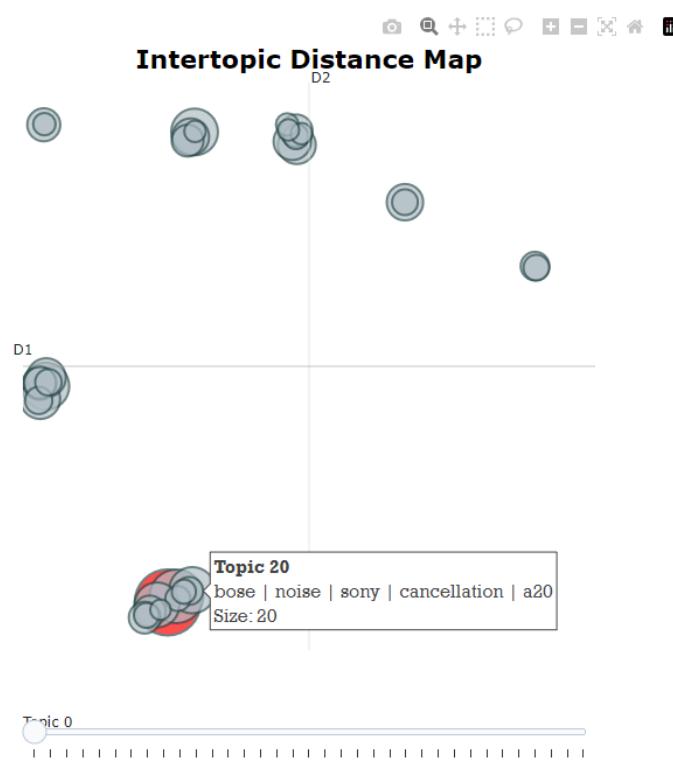
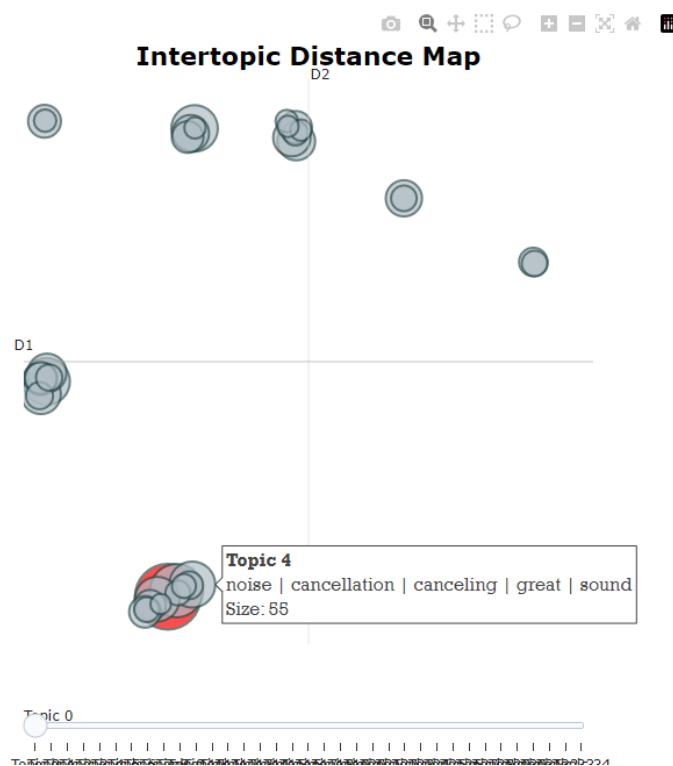
Topic	Count	Name	Representation	Representative_Docs
0	-1	402 -1_headphone_sound_like_good	[headphone, sound, like, good, bass, ear, real...]	[preface review stating reviewed great number ...]
1	0	112 0_headphone_sound_bass_like	[headphone, sound, bass, like, music, good, ge...]	[Updated October 7th, 2022:These still best so...]
2	1	74 1_bose_sony_sound_noise	[bose, sony, sound, noise, better, headphone, ...]	[Around holidays, decided treat pair high qual...]
3	2	57 2_bluetooth_sound_use_noise	[bluetooth, sound, use, noise, device, well, p...]	[Basically, box loved almost everything BT-600...]
4	3	56 3_de_que_el_para	[de, que, el, para, los, la, se, muy, en, sonido]	[Son impecables estos audífonos, su diseño, su...]
5	4	55 4_noise_cancellation_canceling_great	[noise, cancellation, canceling, great, sound,...]	[sound quality really good noise cancellation ...]
6	5	52 5_headphone_sound_cable_ear	[headphone, sound, cable, ear, head, like, hea...]	[Basic Build:Obviously, build appearance headp...]
7	6	41 6_headphone_bass_anc_sound	[headphone, bass, anc, sound, work, battery, h...]	[Love EDM?! Body vibrating BASS?! Want feeeee!..]
8	7	39 7_tv_volume_hear_watching	[tv, volume, hear, watching, hearing, loud, wi...]	[great people can't hear good anymore. grandma...]
9	8	37 8_child_school_small_ear	[child, school, small, ear, situation, one, fi...]	[use iPads, phone device little kid days, they...]
10	9	36 9_headphone_good_great_comfortable	[headphone, good, great, comfortable, sound, q...]	[house become headset house gaming work home. ...]
11	10	35 10_comfortable_wear_price_great	[comfortable, wear, price, great, sound, e7, e...]	[buy E7 half price year ago. It's really great..]
12	11	34 11_price_product_love_thanks	[price, product, love, thanks, priced, excelle...]	[Good product price, Love price n product, Lov...]
13	12	28 12_headset_gaming_latency_wireless	[headset, gaming, latency, wireless, game, sku...]	[Skullcandy PLYR similar design SteelSeries Ar...]
14	13	28 13_bass_mic_teams_hear	[bass, mic, teams, hear, zoom, headset, meetin...]	[working perfectly fine Zoom quite comfortable...]
15	14	27 14_love_battery_headphones_life	[love, battery, headphones, life, good, length...]	[Solid construction comfortable (would add sta...]
16	15	26 15_charge_pair_bose_work	[charge, pair, bose, work, email, phone, day, ...]	[bought used pair Amazon nice discount. Came m...]
17	16	26 16_noise_ear_good_fit	[noise, ear, good, fit, great, quality, clear,...]	[First let tell audiophile max, picky sound qu...]
18	17	26 17_sf200s_year_years_ive	[sf200s, year, years, ive, headphone, last, im...]	[Here's problem headphones: simply last long. ...]
19	18	26 18_pair_christmas_gift_them	[pair, christmas, gift, them, year, second, or...]	[ordered daughter Christmas gift replace 3 yea...]
20	19	21 19_kid_working_child_seller	[kid, working, child, seller, son, children, w...]	[realize meant child probably lasted long got ...]
21	20	20 20_bose_noise_sony_cancellation	[bose, noise, sony, cancellation, a20, better,...]	[frequent flier, depend heavily noise cancelli...]
22	21	19 21_headphone_uncomfortable_use_work	[headphone, uncomfortable, use, work, sound, p...]	[headphone guitar practice. even used I'm alre...]
23	22	18 22_headset_jabra_mute_evolv	[headset, jabra, mute, evolv, work, mic, call...]	[using Jabra Evolve 65 UC headset every day ne...]
24	23	17 23_broke_came_product_daughter	[broke, came, product, daughter, replacementpl...]	[bought son daughter school. son broke within ...]
25	24	17 24_described_reasonable_fast_old	[described, reasonable, fast, old, loved, reli...]	[old one going bad. I'm glad found one like ol...]
26	25	16 25_ear_apple_studio3_beats	[ear, apple, studio3, beats, airpods, like, wo...]	[Going start disagreeing tango's 1-star review..]
27	26	16 26_good_quality_pin_mower	[good, quality, pin, mower, lawn, volume, auto...]	[like everything them. took minute figure use ...]
28	27	16 27_pair_warranty_would_issue	[pair, warranty, would, issue, headphone, repl...]	[I've two major problem Monoprice headphonesTh...]
29	28	15 28_anc_audio_sennheiser_headphone	[anc, audio, sennheiser, headphone, sound, 350...]	[sennheiser HD450 BT successor 4.50. improveme...]
30	29	13 29_vankyo_c750_price_headphone	[vankyo, c750, price, headphone, bass, dollar,...]	[first thing first—for price headphone excepti...]

Στην επόμενη εικόνα φαίνονται οι 10 καλύτεροι όροι του πρώτου topic μαζί με την σχετική σημαντικότητα τους.

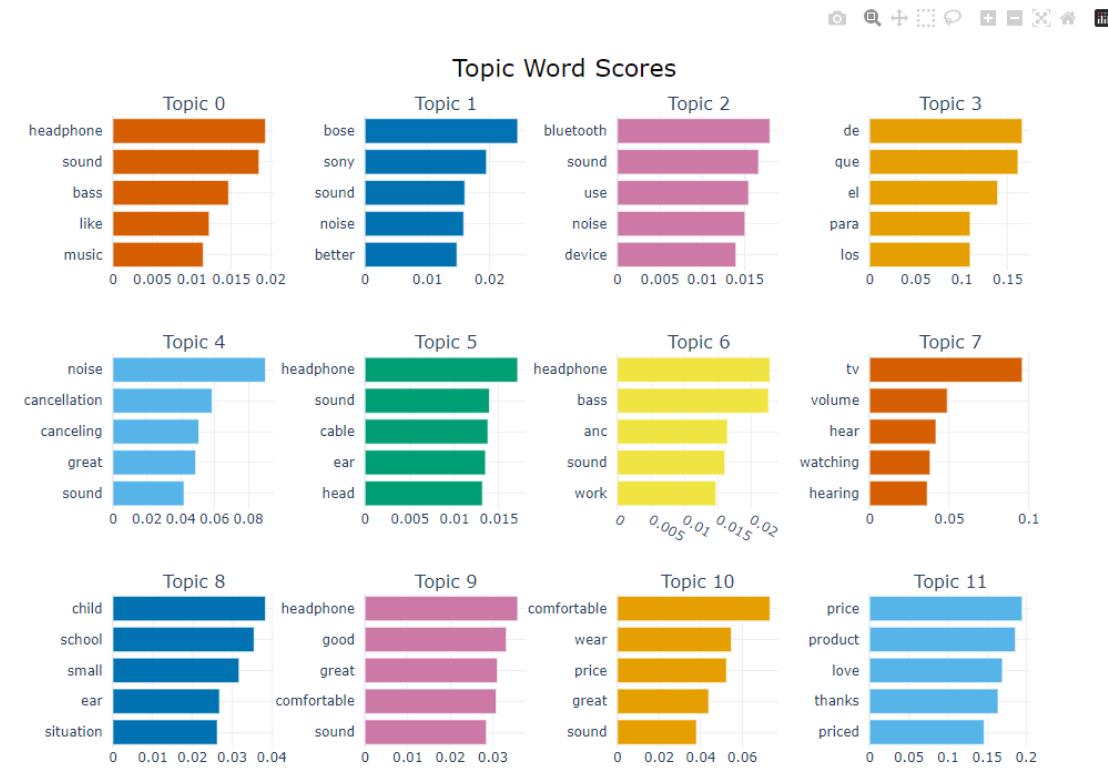
```
[('headphone', 0.05957002185390211),
 ('sound', 0.05741579622023366),
 ('like', 0.03282456214840563),
 ('good', 0.031388893756745355),
 ('quality', 0.030104001513523135),
 ('ear', 0.028457702048972376),
 ('noise', 0.025534242076391538),
 ('bass', 0.02483668396119136),
 ('music', 0.02475826003054612),
 ('use', 0.024302246975419325)]
```

Οι σχέσεις μεταξύ των topics μπορούν να φανούν με τον Intertopic Distance Map ο οποίος μετρά την διαφορά μεταξύ των topics. Τα πιο όμοια είναι πιο κοντά και η απόσταση μεγαλώνει όσο διαφοροποιείται το ένα με το άλλο.

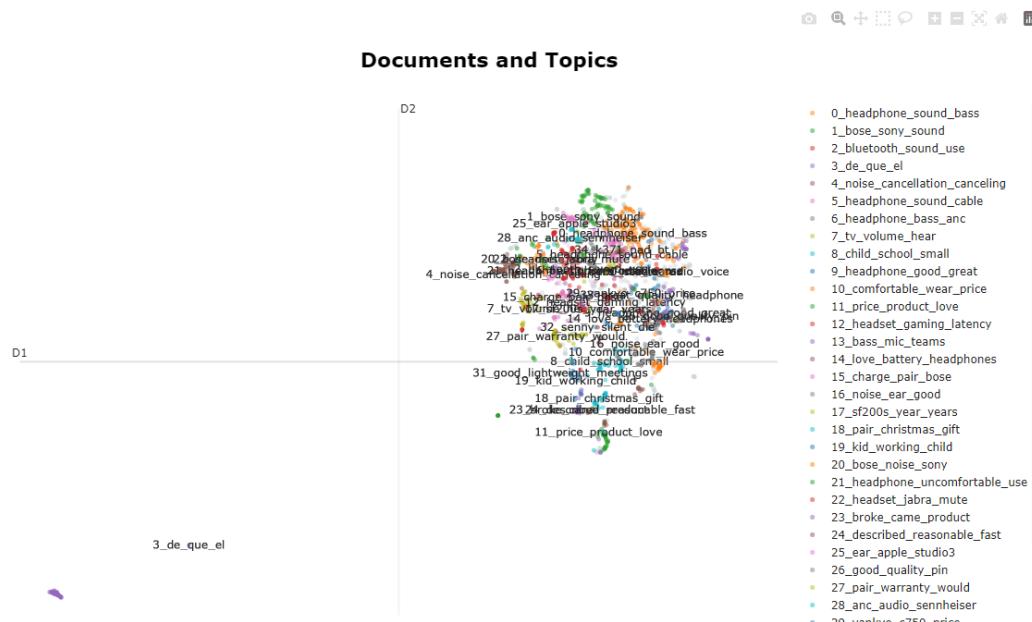
Το μέγεθος του κύκλου είναι ο αριθμός documents μέσα σε κάθε topic.



Άλλος τρόπος οπτικοποίησης αποτελεσμάτων είναι τα bar charts για κάθε αναπαράσταση θέματος.



Τέλος γίνεται οπτικοποίηση των document embeddings σε 2 διαστάσεις για πιο λεπτομερή αποτελέσματα.



## 5.5. Ανάλυση Συνναισθήματος με Μηχανική Μάθηση

Πέραν του Topic Modelling υλοποιήθηκαν διάφοροι βασικοί Machine Learning αλγόριθμοι για την σύγκρισή τους με το επόμενο θέμα, BERT. Παρακάτω θα αναπτυχθεί η υλοποίηση του αναγκαίου preprocessing των δεδομένων, του αλγόριθμου Logistic Regression, Decision Tree Classifier, Bagging Classifier, Random Forest Classifier και Support Vector Classifier.

### 5.5.1. Word2Vec

Όπως και πριν, αλλά και γενικότερα στους αλγορίθμους μηχανικής μάθησης, το πρώτο βήμα είναι η προεπεξεργασία των δεδομένων. Εξεκινώντας λοιπόν από με την προεπεξεργασία των δεδομένων με την χρήση της μεθόδου word2vec της gensim διανυσματοποιούνται τα reviews. Υλοποιήθηκε και η απλή μέθοδος countVectorizer της SkLearn αλλά δεν προτιμήθηκε λόγω χαμηλότερης απόδοσης.

To Word2vec είναι μια ομάδα σχετικών μοντέλων που χρησιμοποιούνται για την παραγωγή word embeddings. Αυτά τα μοντέλα είναι shallow, 2 επιπέδων νευρωνικά δίκτυα τα οποία είναι εκπαιδευμένα για να ανασυνθέτουν γλωσσικά πλαίσια λέξεων. To Word2vec λαμβάνει ως είσοδο ένα μεγάλο σώμα κειμένου και παράγει ένα διανυσματικό χώρο, συνήθως αρκετών εκατοντάδων διαστάσεων, με κάθε μοναδική λέξη στο σώμα να έχει ένα αντίστοιχο διάνυσμα στο διάστημα.

Για την υλοποίηση του είναι απαραίτητη η ύπαρξη κάποιου λεξικού ώστε να δημιουργηθούν τα διανύσματα. Χρησιμοποιείται το προ-εκπαιδευμένο λεξικό της gensim και φορτώνεται για την χρήση του. Κάθε προ-επεξεργασμένη πρόταση (δίχως stopwords, δίχως special characters και lowercased) γίνεται tokenized και κάθε token προστίθεται σαν μέση τιμή της αντίστοιχης που βρέθηκε στο λεξικό ώστε να δημιουργεί ένα διάνυσμα για κάθε πρόταση. Κάθε λέξη που δεν υπάρχει στο λεξικό παίρνει την τιμή 0. Όταν είναι έτοιμο το dataframe με τα διανύσματα γίνεται transpose ώστε να δημιουργηθούν οι σωστές διαστάσεις που χρειάζονται για όρισμα του στις μεθόδους Μηχανικής Μάθησης που θα ακολουθήσουν.

```
model = Word2Vec.load('readyvocab.model')      #reads the vocabulary

processed_sentences = []
for sentence in df['processed']:
    processed_sentences.append(gensim.utils.simple_preprocess(sentence))      #for every sentence in tweets tokenizes each words

vectors = {}
i = 0
for v in processed_sentences:
    vectors[str(i)] = []
    for k in v:
        try:
            vectors[str(i)].append(model.wv[k].mean())      #appends the vector of the word
        except:
            vectors[str(i)].append(np.nan)      #if the word doesnt exist the vocabulary insert it as a Nan value
    i += 1

df_input = pd.DataFrame(dict([(k, pd.Series(v)) for k, v in vectors.items()]))      #puts the vectors in a dataframe
df_input.fillna(value=0.0, inplace=True)      #replace Nan values with 0

df_input = df_input.transpose()      #transposes the matrices in order to insert into the models
```

Εικόνα 5-6: Snippet κώδικα για τη χρήση των αλγορίθμων μηχανικής μάθησης Word2vec.

### 5.5.2. Διαμέριση δοκιμαστικών, ελεγκτικών δεδομένων

Για την χρήση των αλγορίθμων μηχανικής μάθησης που θα υλοποιηθούν χρειάζεται να υπάρχει ένα train dataset το οποίο θα εκπαιδεύσει τους αλγορίθμους καθώς και ένα test dataset το οποίο γνωρίζοντας τα αποτελέσματα θα συγκρίνει τις προβλέψεις και θα βγάζει τις μετρικές ακρίβειας. Για την δημιουργία αυτών των dataset χωρίζεται το προ-υπάρχον dataset σε 2 μέρη με το 70% των δεδομένων του ως train dataset και το υπόλοιπο 30% ως test dataset. Για την υλοποίηση αυτού του split χρησιμοποιήθηκε η πολύ γνωστή μέθοδος train\_test\_split της sklearn βιβλιοθήκης.

### 5.5.3. Hyperparameter Tuning

Στο μηχανική μάθηση, το hyperparameter tuning είναι το πρόβλημα της επιλογής ενός συνόλου βέλτιστων hyperparameters για έναν αλγόριθμο εκμάθησης. Μια hyperparameter είναι μια παράμετρος της οποίας η τιμή χρησιμοποιείται για τον έλεγχο της διαδικασίας μάθησης των αλγορίθμων (learning process). Το ίδιο είδος μοντέλου μηχανικής μάθησης μπορεί να απαιτεί διαφορετικούς περιορισμούς, βάρη ή ρυθμούς εκμάθησης για τη γενίκευση διαφορετικών μοτίβων δεδομένων. Αυτές οι μετρικές ονομάζονται hyperparameters και πρέπει να ρυθμιστούν έτσι ώστε το μοντέλο να μπορεί να λύσει βέλτιστα το πρόβλημα μηχανικής εκμάθησης. Το hyperparameter tuning βρίσκει μια πλειάδα hyperparameters που αποδίδει ένα βέλτιστο μοντέλο που ελαχιστοποιεί μια προκαθορισμένη συνάρτηση απώλειας σε δεδομένα ανεξάρτητα δεδομένα. Η αντικειμενική συνάρτηση παίρνει μια πλειάδα hyperparameters και επιστρέφει τη σχετική απώλεια. Η διασταυρούμενη επικύρωση χρησιμοποιείται συχνά για την εκτίμηση αυτής της απόδοσης γενίκευσης, και επομένως επιλέγεται το σύνολο τιμών για τις hyperparameters που τη μεγιστοποιούν.

Για τους σκοπούς της διεκπεραίωσης της παρούσας διπλωματικής εργασίας, για την εύρεση των βέλτιστων hyperparameters για κάθε μέθοδο χρησιμοποιήθηκε η μέθοδος RandomizedSearchCV και η GridSearchCV της SkLearn.

### 5.5.4. Μετρικές Ακρίβειας

Οι μετρικές ακρίβειας είναι μετρήσεις που χρησιμοποιούνται για να αξιολογήσουν την απόδοση ενός μοντέλου μηχανικής μάθησης. Αντιπροσωπεύουν το πόσο καλά το μοντέλο προβλέπει τις σωστές τιμές ή την κατηγορία των δεδομένων εισόδου. Οι μετρικές ακρίβειας εξαρτώνται από τον τύπο του προβλήματος (π.χ. ταξινόμηση, παλινδρόμηση) και τον τρόπο που το μοντέλο εκπαιδεύτηκε.

Ορισμένες συνηθισμένες μετρικές ακρίβειας περιλαμβάνουν:

- **Ακρίβεια (Accuracy):** Είναι ο ποσοστός των σωστών προβλέψεων συνολικά. Υπολογίζεται ως τον αριθμό των σωστών προβλέψεων διαιρούμενο με τον συνολικό αριθμό παραδειγμάτων.
- **Ακρίβεια Κλάσης (Class Accuracy):** Αναφέρεται στην ακρίβεια για κάθε κατηγορία ξεχωριστά σε ένα πρόβλημα ταξινόμησης. Υπολογίζεται ως τον αριθμό των σωστών προβλέψεων για μια συγκεκριμένη κατηγορία διαιρούμενο με τον συνολικό αριθμό παραδειγμάτων αυτής της κατηγορίας.
- **Ανάκληση (Recall):** Αναφέρεται στην ικανότητα του μοντέλου να εντοπίσει όλες τις σωστές περιπτώσεις μιας κατηγορίας. Υπολογίζεται ως τον αριθμό των σωστών

προβλέψεων για μια κατηγορία διαιρούμενο με τον συνολικό αριθμό πραγματικών παραδειγμάτων αυτής της κατηγορίας.

- Ακρίβεια Πρόβλεψης (Precision): Αναφέρεται στην ικανότητα του μοντέλου να προβλέπει σωστά μια συγκεκριμένη κατηγορία. Υπολογίζεται ως τον αριθμό των σωστών προβλέψεων για μια κατηγορία διαιρούμενο με τον συνολικό αριθμό προβλέψεων για αυτή την κατηγορία.

Αυτές είναι μερικές από τις πιο συνηθισμένες μετρικές ακρίβειας, αλλά υπάρχουν και πολλές άλλες, όπως η F1-score, η καμπύλη ROC κ.ά., που χρησιμοποιούνται για την αξιολόγηση και σύγκριση μοντέλων μηχανικής μάθησης. Η επιλογή της κατάλληλης μετρικής εξαρτάται από το συγκεκριμένο πρόβλημα και τις απαιτήσεις του.

Για τους σκοπούς της παρούσας διπλωματικής εργασίας και δεδομένου του τύπου προβλήματος μηχανικής μάθησης που επιδιώκουμε να επιλύσουμε, για κάθε μέθοδο μηχανικής μάθησης που θα ακολουθήσει πάρθηκαν οι τιμές ακρίβειας σύμφωνα με τις μετρικές, accuracy, precision και recall score.

- (1) Το *accuracy score* υπολογίζει τη βαθμολογία ακρίβειας για ένα σύνολο προβλεπόμενων ετικετών έναντι των αληθινών ετικετών.
- (2) Το *precision score* είναι ο λόγος  $tp / (tp + fp)$  όπου  $tp$  είναι ο αριθμός των αληθινών θετικών και  $fp$  ο αριθμός των ψευδών θετικών. Η ακρίβεια είναι διαισθητικά η ικανότητα του ταξινομητή να μην επισημαίνει ως θετικό ένα δείγμα που είναι αρνητικό. Η καλύτερη τιμή είναι 1 και η χειρότερη τιμή είναι 0.
- (3) Το *recall score* είναι ο λόγος  $tp / (tp + fn)$  όπου  $tp$  είναι ο αριθμός των αληθινών θετικών και  $fn$  ο αριθμός των ψευδών αρνητικών. Η ανάκληση είναι διαισθητικά η ικανότητα του ταξινομητή να βρίσκει όλα τα θετικά δείγματα. Η καλύτερη τιμή είναι 1 και η χειρότερη τιμή είναι 0.

### 5.5.5. Logistic Regression

Αυτός ο τύπος στατιστικού μοντέλου (γνωστό και ως μοντέλο logit) χρησιμοποιείται συχνά για ταξινόμηση και προγνωστική ανάλυση. Η λογιστική παλινδρόμηση εκτιμά την πιθανότητα να συμβεί ένα συμβάν, όπως ψήφισαν ή δεν ψήφισαν, με βάση ένα δεδομένο σύνολο ανεξάρτητων μεταβλητών.

Για την υλοποίηση του αλγορίθμου Logistic Regression χρησιμοποιήθηκε η μέθοδος LogisticRegression της SkLearn με βέλτιστα hyperparameters, solver: liblinear και penalty: l2. Παρακάτω φαίνονται τα αποτελέσματα των μετρικών.

```
accuracy_score: 0.5045454545454545
precision_score: [0.          0.          1.          0.42857143  0.52124646]
recall_score:  [0.          0.          0.03703704  0.25531915  0.83257919]
```

Εικόνα 5-7: Μετρικές αλγορίθμου Logistic Regression.

### 5.5.6. Decision Tree Classifier

Τα Δέντρα Αποφάσεων είναι μια μη παραμετρική εποπτευόμενη μέθοδος μάθησης που χρησιμοποιείται για ταξινόμηση και παλινδρόμηση. Ο στόχος είναι να δημιουργηθεί ένα

μοντέλο που προβλέπει την τιμή μιας μεταβλητής στόχου μαθαίνοντας απλούς κανόνες απόφασης που συνάγονται από τα χαρακτηριστικά δεδομένων. Ένα δέντρο μπορεί να θεωρηθεί ως μια τμηματικά σταθερή προσέγγιση.

Για την υλοποίηση του Decision Tree Classifier χρησιμοποιήθηκε η μέθοδος DecisionTreeClassifier της SkLearn. Δημιουργήθηκαν 5 διαφορετικά μοντέλα με διαφορετικές hyperparameters και καταγράφηκε η μέση απόδοση των μοντέλων αυτών. Για την δημιουργία των 5 διαφορετικών μοντέλων χρησιμοποιήθηκε η μέθοδος cross\_val\_score και η μέση τιμή των μετρικών φαίνεται παρακάτω:

Mean DTC score: 0.4878048780487805

#### 5.5.7. Bagging Classifier

Ο Bagging Classifier είναι ένας μετα-εκτιμητής συνόλου που βάζει τους βασικούς ταξινομητές σε τυχαία υποσύνολα του αρχικού συνόλου και στη συνέχεια συγκεντρώνει τις μεμονωμένες προβλέψεις τους (είτε με ψηφοφορία είτε με μέσο όρο) για να σχηματίσει μια τελική πρόβλεψη. Ένας τέτοιος meta-estimator μπορεί τυπικά να χρησιμοποιηθεί ως τρόπος μείωσης της διακύμανσης ενός black box classifier (π.χ. Decision Tree), εισάγοντας την τυχαιοποίηση στη διαδικασία κατασκευής του και στη συνέχεια δημιουργώντας ένα σύνολο από αυτό.

Για την υλοποίηση του χρησιμοποιήθηκε η μέθοδος BaggingClassifier της SkLearn βιβλιοθήκης και το αποτέλεσμα των μετρικών με και χωρίς cross\_val\_score φαίνεται παρακάτω:

Bagging score: 0.5113636363636364  
Mean Bagging score: 0.5678048780487805

#### 5.5.8. Random Forest Classifier

Ο Random Forest Trees είναι ένας αλγόριθμος μηχανικής μάθησης που βασίζεται σε δέντρα αποφάσεων. Τα τυχαία δέντρα ανήκουν σε μια κατηγορία αλγορίθμων μηχανικής μάθησης που κάνει ταξινόμηση συνόλου. Ο όρος σύνολο υποδηλώνει μια μέθοδο που κάνει προβλέψεις με τη λήψη μέσου όρου σε σχέση με τις προβλέψεις πολλών ανεξάρτητων βασικών μοντέλων.

Για την υλοποίηση του χρησιμοποιήθηκε η μέθοδος RandomForestClassifier της SkLearn και το αποτέλεσμα των μετρικών με και χωρίς cross\_val\_score φαίνεται παρακάτω:

Random Forest Classifier:  
Mean Random Forest score: 0.5795121951219512

Προς έκπληξη, ο Random Forest Classifier έδειξε το υψηλότερο score από τις υπόλοιπες μεθόδους. Ο Random Forest Classifier είναι ένας δημοφιλής αλγόριθμος μηχανικής μάθησης που χρησιμοποιείται για προβλήματα ταξινόμησης. Είναι ένα σύνολο από απλά δέντρα

αποφάσεων (decision trees), τα οποία εκπαιδεύονται με τη μέθοδο του συνόλου (ensemble learning). Κάθε δέντρο στο σύνολο λαμβάνει τυχαία δείγματα δεδομένων και χαρακτηριστικών, με την αρχή του "τυχαιοποιημένου δάσους" (random forest).

Ο Random Forest Classifier έχει αρκετά πλεονεκτήματα. Ένα από αυτά είναι η ικανότητά του να αντιμετωπίζει αποτελεσματικά προβλήματα με μεγάλο αριθμό χαρακτηριστικών, ενώ παραμένει ανθεκτικός σε υπερπροσαρμογή. Επίσης, ο Random Forest είναι σε θέση να αντιμετωπίσει αποτελεσματικά την ανισορροπία κλάσεων σε ένα σύνολο δεδομένων.

### 5.5.9. Support Vector Classifier

Ο SVC λειτουργεί αντιστοιχίζοντας δεδομένα σε ένα χώρο χαρακτηριστικών υψηλών διαστάσεων, έτσι ώστε τα σημεία δεδομένων να μπορούν να κατηγοριοποιηθούν, ακόμη και όταν τα δεδομένα δεν είναι γραμμικά διαχωριζόμενα. Βρίσκεται ένας διαχωριστής μεταξύ των κατηγοριών και στη συνέχεια τα δεδομένα μετασχηματίζονται με τέτοιο τρόπο ώστε το διαχωριστικό να μπορεί να σχεδιαστεί ως υπερεπίπεδο.

Για την υλοποίηση του χρησιμοποιήθηκε η μέθοδος SVC της SkLearn. Αποτελέσματα δίχως hyperparameter tuning φαίνονται παρακάτω:

```
accuracy_score: 0.5159090909090909
precision_score: [0.          0.          1.          0.57142857 0.5106383 ]
recall_score:  [0.          0.          0.05555556 0.05673759 0.97737557]
```

Μετά το hyperparameter tuning και βρίσκοντας ως βέλτιστες hyperparameters τα C: 10, gamma: 1 και kernel: RBF οι μετρικές φαίνονται παρακάτω:

```
accuracy_score: 0.5522727272727272
precision_score: [0.          1.          0.46153846 0.53947368 0.55331412]
recall_score:  [0.          0.23529412 0.11111111 0.29078014 0.86877828]
```

### 5.5.10. BERT

Η τεχνική topic modelling BERT αναλύθηκε στο Κεφάλαιο 4.2 μαζί με ένα βήμα-προς-βήμα παράδειγμα. Στο συγκεκριμένο υποκεφάλαιο, θα περιγράψουμε πως αυτή η τεχνική χρησιμοποιήθηκε για τους σκοπούς διεκπεραίωσης της παρούσας διπλωματικής εργασίας.

Όπως έχουμε προαναφέρει, ο αλγόριθμος BERT είναι μια μέθοδος προεκπαίδευσης γλωσσικών αναπαραστάσεων. Η προεκπαίδευση αναφέρεται στον τρόπο με τον οποίο ο BERT εκπαιδεύεται αρχικά σε μια μεγάλη πηγή κειμένου, όπως η Wikipedia. Στη συνέχεια, μπορούμε να εφαρμόσουμε τα αποτελέσματα της εκπαίδευσης σε άλλες εργασίες Επεξεργασίας Φυσικής Γλώσσας (NLP), όπως η απάντηση σε ερωτήσεις και η ανάλυση συναισθημάτων.

Για την χρήση BERT, το dataset σε αντίθεση με άλλες μεθόδους πρέπει να είναι σε αρχική, απείραχτη μορφή, με stopwords καθώς αυτά πολλές φορές καθορίζουν το πραγματικό νόημα της πρότασης και ως αποτέλεσμα και του κειμένου.

Ο αλγόριθμος BERT βασίζεται στους Transformers. Ένας transformer είναι μια αρχιτεκτονική βαθιάς μάθησης που βασίζεται στον μηχανισμό προσοχής και είναι μακράν γρηγορότερο προηγούμενων μεθόδων όπως LSTM (Long Short Term Memory) μοντέλων.

Για την υλοποίηση του αρχικά χωρίζεται σε train και test το dataset, όπως έχουμε εξηγήσει και παραπάνω, και δημιουργείται ένας tokenizer σύμφωνα με τους κανόνες του DistilBert και σύμφωνα με αυτό γίνονται tokenized οι προτάσεις καθώς και κόβονται μεγαλύτερον από του επιτρεπτού μήκους. Μετά δημιουργείται ένα pad σύμφωνα με τον tokenizer. Επιλέγονται μετρικές και τα labels δηλαδή η αξιολόγηση των reviews ονομάζονται.

Στη συνέχεια, το ακατέργαστο dataset μετατρέπεται σε μορφή tf\_dataset, Αυτό επιβάλλεται καθώς ο αλγόριθμος BERT προετοιμάζει και προ-επεξεργάζεται τα ακατέργαστα δεδομένα με την μέθοδο prepare\_df\_dataset().

Τα μοντέλα transformers έχουν όλα μια προεπιλεγμένη λειτουργία απώλειας σχετικής με την εργασία, επομένως δεν χρειάζεται να καθοριστεί κάποια εκτός και αν είναι επιθυμητό. Για τους σκοπούς της παρούσας εργασίας, προχωράμε με finetuning στο μοντέλο και επιλέγονται hyperparameters σύμφωνα με τις παρακάτω εικόνες.

```
from transformers import create_optimizer
import tensorflow as tf

batch_size = 16
num_epochs = 5
batches_per_epoch = len(tokenized_review['train']) //batch_size
total_train_steps = int(batches_per_epoch * num_epochs)
optimizer, schedule = create_optimizer(init_lr=2e-5, num_warmup_steps=0, num_train_steps=total_train_steps)

    tf_train_set = model.prepare_tf_dataset(
        tokenized_review['train'],
        shuffle=False,
        batch_size=16,
        collate_fn=data_collator
    )

    tf_validation_set = model.prepare_tf_dataset(
        tokenized_review["test"],
        shuffle=False,
        batch_size=16,
        collate_fn=data_collator,
    )
```

Εικόνα 5-8: finetuning και επιλογή hyperparameters.

Κάνοντας fit με λίγες epochs επιστρέφονται τα παρακάτω αποτελέσματα. Εδώ επιλέχθηκαν 5 epochs αλλά και 3 ήταν υπέρ αρκετές καθώς, όπως φαίνεται και στην Εικόνα που ακολουθεί, τα transformers συγκλίνουν πολύ σύντομα.

```
Epoch 1/5
6/65 [=>.....] - ETA: 53s - loss: 1.1864WARNING:tensorflow:Callback method `on_train_ba
65/65 [=====] - 74s 1s/step - loss: 1.0532 - val_loss: 1.1107 - accuracy: 0.5023
Epoch 2/5
65/65 [=====] - 72s 1s/step - loss: 0.9825 - val_loss: 1.0972 - accuracy: 0.5205
Epoch 3/5
65/65 [=====] - 74s 1s/step - loss: 0.9626 - val_loss: 1.0972 - accuracy: 0.5205
Epoch 4/5
65/65 [=====] - 72s 1s/step - loss: 0.9597 - val_loss: 1.0972 - accuracy: 0.5205
Epoch 5/5
65/65 [=====] - 74s 1s/step - loss: 0.9624 - val_loss: 1.0972 - accuracy: 0.5205
<keras.callbacks.History at 0x7f19bf528bb0>
```

Εικόνα 5-9: Σύγκλιση epochs.



## 6. Επίλογος

### 6.1. Συμπεράσματα

Η σύγχρονη αγορά προϊόντων έχει εξελιχθεί σε έναν πολύπλοκο και ανταγωνιστικό χώρο, με αμέτρητες επιλογές για τους καταναλωτές. Κατά την αγορά προϊόντων, οι καταναλωτές αντιμετωπίζουν πολλές παραμέτρους που επηρεάζουν τις αποφάσεις τους, όπως η ποιότητα, η τιμή, οι προσωπικές προτιμήσεις και οι συναισθηματικές αντιδράσεις. Συνεπώς, η ικανότητα των καταναλωτών να λαμβάνουν αποφάσεις αγοράς που ανταποκρίνονται στις ατομικές τους ανάγκες και προτιμήσεις είναι ζωτικής σημασίας.

Στόχος αυτής της εργασίας είναι η ανάπτυξη ενός συστήματος υποστήριξης αποφάσεων που βασίζεται στην ανάλυση συναισθημάτων, με σκοπό να παρέχει την κατάλληλη πληροφορία στους καταναλωτές για τη λήψη ενημερωμένων αποφάσεων αγοράς. Αυτό το σύστημα θα τους δώσει τη δυνατότητα να αντιμετωπίζουν τον όγκο των προϊόντων που είναι διαθέσιμα και να επιλέγουν αυτά που θα ικανοποιήσουν καλύτερα τις ανάγκες και τις προτιμήσεις τους και θα προσφέρουν τη μεγαλύτερη ικανοποίηση.

Για να επιτευχθεί αυτός ο στόχος, μελετήσαμε δύο διαφορετικές προσεγγίσεις για την αναγνώριση συναισθημάτων. Η πρώτη προσέγγιση αφορά τη χρήση αλγορίθμων αναγνώρισης συναισθημάτων χωρίς τη χρήση μηχανικής μάθησης. Αυτή η προσέγγιση εξετάζει τη χρήση κανόνων και προκαθορισμένων μοντέλων για την ανίχνευση συναισθημάτων σε κείμενα και άλλα αποτελέσματα. Ωστόσο, καταλήξαμε στο συμπέρασμα ότι η χρήση αλγορίθμων μηχανικής μάθησης προσφέρει καλύτερη απόδοση και ακρίβεια στην αναγνώριση συναισθημάτων. Για την υλοποίηση του συστήματος, χρησιμοποιήσαμε τις βιβλιοθήκες Scikit-learn, TensorFlow και Keras.io. Αυτές οι βιβλιοθήκες προσφέρουν ισχυρά εργαλεία και πλαίσια για την ανάπτυξη αλγορίθμων μηχανικής μάθησης και βαθιάς μάθησης. Χρησιμοποιώντας αυτές τις εργαλειοθήκες, μπορέσαμε να εκπαιδεύσουμε μοντέλα μηχανικής μάθησης που δύνανται να αναγνωρίσουν συναισθήματα από δεδομένα κειμένου, ήχου ή εικόνας. Η επιλογή αυτών των βιβλιοθηκών βασίστηκε στην αξιοπιστία, την ευελιξία και την ευκολία χρήσης που προσφέρουν.

Σε αυτήν την εργασία, ο αναγνώστης θα έχει την ευκαιρία να εξερευνήσει τη διαδικασία ανάπτυξης του συστήματος υποστήριξης αποφάσεων βασισμένου στην ανάλυση συναισθημάτων. Σε κάθε κεφάλαιο, θα παρουσιάσουμε μια συνολική άποψη για το πρόβλημα που λύνουμε και τον σκοπό της εκάστοτε ανάλυσης. Θα εξηγήσουμε τις μεθόδους που χρησιμοποιήσαμε για την αναγνώριση συναισθημάτων και θα παρουσιάσουμε τα αποτελέσματα που προέκυψαν από την εφαρμογή των αλγορίθμων. Τέλος, θα αξιολογήσουμε τα αποτελέσματα και θα συζητήσουμε τις πιθανές επεκτάσεις και βελτιώσεις που μπορούν να γίνουν στο μέλλον. Με την ολοκλήρωση αυτής της εργασίας, αναμένεται ότι ο αναγνώστης θα έχει αποκτήσει μια πλήρη κατανόηση του προβλήματος που αντιμετωπίζουμε και των προτεινόμενων λύσεων που αναπτύξαμε με τη χρήση αλγορίθμων μηχανικής μάθησης.

Όσον αφορά την υλοποίηση ο web scraper δούλεψε άψογα δίχως να υπερφορτώνει το υπολογιστικό σύστημα και οι χρόνοι ολοκλήρωσης του ήταν αποδεκτοί. Τα dataset που δημιουργούσε ήταν καθαρά χωρίς ανάγκη για πολύ επεξεργασία. Μόνη αδυναμία του η περίπτωση που χαθεί η σύνδεση καθώς δεν υπάρχει χρήση κάποιου rollback, η ελλιπής εγγραφή κάποιων δεδομένων. Το preprocessing που χρειάστηκε ήταν αρκετά απλό καθώς τα δεδομένα ήταν εξ αρχής καθαρά. Γλώσσες πέραν των Αγγλικών έχει αγνοηθεί. Περνώντας στο Latent Dirichlet Allocation δημιουργείται μια συσχέτιση λέξεων η οποία είναι αρκετά

αναμενόμενη, κάτι που είναι θετικό διότι δείχνει πως τα αποτελέσματα ανταποκρίνονται στον πραγματικό κόσμο. Οι διάφοροι μέθοδοι της SkLearn έδειξαν περίπου ίδια αποτελέσματα και το accuracy να κυμαίνεται γύρω στο 0.6, τιμή αναμενόμενη από το μέγεθος του dataset. Τέλος η υλοποίηση BERT με το πλεονέκτημα χρόνου υλοποίησης καθώς και την μη χρήση preprocessed δεδομένων έδειξε τα μεγαλύτερα αποτελέσματα accuracy και είναι μακράν ο προτεινόμενος αλγόριθμος πρόβλεψης συναισθημάτων σε dataset κειμένων.

## 6.2. Μελλοντικές επεκτάσεις

Έχοντας αναπτύξει μια αποτελεσματική μέθοδο πρόβλεψης αξιολογήσεων βασιζόμενη στις αναθεωρήσεις των χρηστών, ως μελλοντική επέκταση σκοπεύουμε να χρησιμοποιήσουμε αυτήν τη μέθοδο για να βελτιώσουμε την ταξινόμηση των αποτελεσμάτων μιας αναζήτησης.

Συνήθως, όταν ο χρήστης αναζητά ένα προϊόν, η ταξινόμηση των αποτελεσμάτων βασίζεται στον αριθμό των αξιολογήσεων ή στο σύνολο των θετικών αξιολογήσεων. Με το σύστημα που έχει αναπτυχθεί, παρέχονται αποτελέσματα που αντανακλούν πιο ακριβώς τις πραγματικές αξιολογήσεις των χρηστών και τα συναισθήματα που προκύπτουν από αυτές.

Με την εφαρμογή του συστήματος, οι ταξινομήσεις σε μια αναζήτηση μπορούν να προκύπτουν από την αξιολόγηση των προϊόντων βάσει των πραγματικών αξιολογήσεων των χρηστών και την αντίληψη των συναισθημάτων που αποτυπώνονται σε αυτές.

Για αυτό τον λόγο δημιουργήθηκε και dataframe το οποίο περιέχει και το κάθε προϊόν μαζί με τις αξιολογήσεις του. Μελλοντική επέκταση, η χρήση του συστήματος που αναφέρθηκε ως συμπληρωματική μέθοδος ταξινόμησης σε μια αναζήτηση.

## 7. Βιβλιογραφία

- Aburomman, A. A., & Ibne Reaz, M. Bin. (2016). A novel SVM-kNN-PSO ensemble method for intrusion detection system. *Applied Soft Computing Journal*, 38, 360–372. <https://doi.org/10.1016/j.asoc.2015.10.011>
- Acien, A., Morales, A., Fierrez, J., Vera-Rodriguez, R., & Bartolome, I. (2020). BeCAPTCHA: Detecting human behavior in smartphone interaction using multiple inbuilt sensors. *ArXiv Preprint ArXiv:2002.00918*. Retrieved from <http://arxiv.org/abs/2002.00918>
- Bishop, C. M. (2006). Pattern Recognition and Machine Learning. In *EAI/Springer Innovations in Communication and Computing*. [https://doi.org/10.1007/978-3-030-57077-4\\_11](https://doi.org/10.1007/978-3-030-57077-4_11)
- Black Duck Open Hub. (2023). Accord.NET Framework. Retrieved June 16, 2023, from <https://openhub.net/p/Accord-NET>
- Gashler, M. (2011). Waffles: A machine learning toolkit. *Journal of Machine Learning Research*, 12, 2383–2387.
- Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media.
- Google. (2023). Machine Learning - Google Colab. Retrieved June 16, 2023, from <https://colab.research.google.com/github/jakevdp/PythonDataScienceHandbook/blob/master/notebooks/05.00-Machine-Learning.ipynb>
- Holmes, G., Donkin, A., & Witten, I. H. (1994). Weka: A machine learning workbench. *Proceedings of the Second Australia and New Zealand Conference on Intelligent Information Systems*. Brisbane, Australia.
- Hu, J., Niu, H., Carrasco, J., Lennox, B., & Arvin, F. (2020). Voronoi-Based Multi-Robot Autonomous Exploration in Unknown Environments via Deep Reinforcement Learning. *IEEE Transactions on Vehicular Technology*, 69(12), 14413–14423. <https://doi.org/10.1109/tvt.2020.3034800>
- IBM. (2021). What is Machine Learning? Retrieved June 15, 2023, from [www.ibm.com](http://www.ibm.com)
- Iliadis, L. S. (2005). A decision support system applying an integrated fuzzy model for long-term forest fire risk estimation. *Environmental Modelling and Software*, 20(5), 613–621. <https://doi.org/10.1016/j.envsoft.2004.03.006>
- Kecman, V. (2001). *Learning and soft computing: support vector machines, neural networks, and fuzzy logic models*. MIT press.
- keras.io. (2023). Why use Keras? Retrieved June 16, 2023, from <https://keras.io/why-use-keras/>
- keras.io. (2023). Core - Keras Documentation. Retrieved June 16, 2023, from <https://keras.io/layers/core/>
- Kotsiantis, S. B. (2013). Decision trees: A recent overview. *Artificial Intelligence Review*, 39(4), 261–283. <https://doi.org/10.1007/s10462-011-9272-4>
- Koza, J. R., Bennett, F. H., Andre, D., & Keane, M. A. (1996). Automated Design of Both the Topology and Sizing of Analog Electrical Circuits Using Genetic Programming. In *Artificial Intelligence in Design '96*. Retrieved from [https://doi.org/10.1007/978-94-009-0279-4\\_9](https://doi.org/10.1007/978-94-009-0279-4_9)
- Liu, F. T., Ting, K. M., & Zhou, Z. H. (2012). Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data*, 6(1).

<https://doi.org/10.1145/2133360.2133363>

- Mohsen, Y.-N., Hugh, E., Tulpan, D., Sulik, J., & Eskandari, M. (2021). Application of Machine Learning Algorithms in Plant Breeding: Predicting Yield From Hyperspectral Reflectance in Soybean? *Front. Plant Sci.*, 11, 624273. <https://doi.org/10.3389/fpls.2020.624273>
- PyTorch. (2023). PyTorch Machine Learning in Python. Retrieved June 15, 2023, from [https://pytorch.org/vision/stable/auto\\_examples/index.html](https://pytorch.org/vision/stable/auto_examples/index.html)
- Reutemann, P., Pfahringer, B., & Frank, E. (2004). Proper: A Toolbox for Learning from Relational Data with Propositional and Multi-Instance Learners. *17th Australian Joint Conference on Artificial Intelligence (AI2004)*. <https://doi.org/10.1.1.459.8443>
- Rivera, E., Tengana, L., Solano, J., Castelblanco, A., López, C., & Ochoa, M. (2020). Risk-based Authentication Based on Network Latency Profiling. *AISec 2020 - Proceedings of the 13th ACM Workshop on Artificial Intelligence and Security*, 105–115. <https://doi.org/10.1145/3411508.3421377>
- Sarangi, S., Sahidullah, M., & Saha, G. (2020). Optimization of data-driven filterbank for automatic speaker verification. *Digital Signal Processing*, 104, 102795. <https://doi.org/https://doi.org/10.1016/j.dsp.2020.102795>
- scikit-learn developers. (2023). scikit-learn Machine Learning in Python. Retrieved June 15, 2023, from [https://scikit-learn.org/stable/auto\\_examples/index.html](https://scikit-learn.org/stable/auto_examples/index.html)
- Shu, L., Xie, J., Yang, M., Li, Z., Li, Z., Liao, D., ... Yang, X. (2018). A review of emotion recognition using physiological signals. *Sensors (Switzerland)*, 18(7). <https://doi.org/10.3390/s18072074>
- Sonnenburg, S., Rätsch, G., Henschel, S., Widmer, C., Behr, J., Zien, A., ... Franc, V. (2010). The Shogun machine learning toolbox. *Journal of Machine Learning Research*, 11, 1799–1802.
- Souza, C. (2017). Machine learning, computer vision, statistics and general scientific computing for .NET: Accord-net/framework. Retrieved June 16, 2023, from <https://github.com/accord-net/framework/blob/development/Release%20notes.txt>
- TensorFlow. (2023a). TensorFlow Machine learning. Retrieved from <https://dzone.com/articles/tensorflow-simplified-examples?fromrel=true>
- TensorFlow. (2023b). Using TPUs | TensorFlow. Retrieved June 16, 2023, from [https://www.tensorflow.org/guide/using\\_tpu](https://www.tensorflow.org/guide/using_tpu)
- Vincent, P., Bengio, Y., Chapados, N., & Delalleau, O. (2023). Plearn high-performance machine learning library. Retrieved June 16, 2023, from <http://plearn.berlios.de/>
- Wikipedia. (2023). Machine learning. Retrieved June 15, 2023, from [https://en.wikipedia.org/wiki/Machine\\_learning#Overview](https://en.wikipedia.org/wiki/Machine_learning#Overview)
- Witten, I., Frank, E., & Hall, M. (2011). Data Mining. In *Encyclopedia of Ecology, Five-Volume Set*. <https://doi.org/10.1016/B978-008045405-4.00153-1>