Computer Science and Information Systems 19(3):1155–1175 https://doi.org/10.2298/CSIS220212034S

Effective Methods for Email Classification: Is it a Business or Personal Email?

Milena Šošić and Jelena Graovac

Faculty of Mathematics, University of Belgrade
Studentski Trg 16, 11000 Belgrade, Serbia
pd202030@alas.matf.bg.ac.rs
jgraovac@matf.bg.ac.rs

Abstract. With the steady increase in the number of Internet users, email remains the most popular and extensively used communication means. Therefore, email management is an important and growing problem for individuals and organizations. In this paper, we deal with the classification of emails into two main categories, Business and Personal. To find the best performing solution for this problem, a comprehensive set of experiments has been conducted with the deep learning algorithms: Bidirectional Long-Short Term Memory (BiLSTM) and Attention-based BiLSTM (BiLSTM+Att), together with traditional Machine Learning (ML) algorithms: Stochastic Gradient Descent (SGD) optimization applied on Support Vector Machine (SVM) and Extremely Randomized Trees (ERT) ensemble method. The variations of individual email and conversational email thread arc representations have been explored to reach the best classification generalization on the selected task. A special contribution of this paper is the extraction of a large number of additional lexical, conversational, expressional, emotional, and moral features, which proved very useful for differentiation between personal and official written conversations. The experiments were performed on the publicly available Enron email benchmark corpora on which we obtained the State-Of-the-Art (SOA) results. As part of the submission, we have made our work publicly available to the scientific community for research purposes.

Keywords: Email classification, business, personal, deep learning, BiLSTM, SGD, BERT embeddings, Tf-Idf, lexicons, NLP.

1. Introduction

In the last decade, emails have become one of the crucial media for both personal and business communication. Despite the rise of social media and instant messaging, email usage is steadily growing, with more than 4 billion users worldwide in 2021 and about 6.8 billion email accounts – and it continues to grow [21]. This is mainly due to their efficiency, low cost, and compatibility with diversified types of information. In the last decade, emails have become one of the crucial media for both personal and business communication. Despite the rise of social media and instant messaging, email usage is steadily growing, with about 6.8 billion email accounts, more than 4 billion users and about 320 billion sent and received emails per day worldwide in 2021 – with expectations for numbers to further increase by 2025 [21]. This is mainly due to their efficiency, low cost, and compatibility with diversified types of information. Observed trend has made the automatic processing of emails more than desirable. For example, the classification of emails into Business and Personal categories can help a lot in better handling the email inbox and decreasing the time spent managing emails every day. This trend has made the automatic processing of emails more than desirable. For example, the classification of emails into Business and Personal categories can help a lot in better handling the email inbox and decreasing the time spent managing emails every day.

To facilitate usage of emails and explore business potentials in emailing, various studies have been proposed such as spam-filtering [17], multi folder classification [25], phishing email classification [1], etc. In this paper, we focus on the classification of emails into two main categories, Business and Personal, which belongs to the text classification task [7], [8].

Unlike other email processing tasks, such as spam filtering, this problem has not received much attention, and it remains a challenging task. One of the reasons for that is a lack of data - personal emails are often highly private, and they are usually unavailable for research purposes. In this study for training and testing purposes, we used two different distributions of the Enron email corpus [14], the sole email corpus that is freely available (public and not licensed).

The main contributions of this paper are as follows:

- Conducting a comprehensive set of experiments using advanced deep learning and traditional machine learning (ML) techniques.
- Experimentation with different variants of individual emails, and conversational thread arcs of emails.
- Experimentation with different text representation techniques on words, word n-grams, character n-grams, and BERT embeddings.
- Extraction of different lexical, conversational, expressional, emotional, and moral features using a diverse set of lexicons and email content characteristics.
- Extensive comparison and evaluation of the obtained results.
- Production of the State-Of-the-Art (SOA) results.

The paper continues with a review of the related work in section 2. It is followed by the presentation of our approach in section 3 including preprocessing, features extraction, and used traditional and deep learning ML techniques. After that, in section 4, the experimental framework is described. The obtained results are expounded in section 5, while section 6 presents the results of comparison with previously published SOA techniques. Section 7 concludes the paper.

We make our work publicly available and reproducible [1].

2. Related Work

Since Enron is the only freely available email data set, many researchers have worked on it with different tasks. To our knowledge, the previous efforts most closely related to our research are [12], [2], and [3]. They all have worked on the same problem: classification of emails into Business or Personal category and used the same Enron data set for training and testing.

[1] https://github.com/milena-sosic/Email-Business-Personal

First attempts to categorize corporate emails into Business and Personal categories were made by [12]. The main contribution of this paper is the largest scale annotation project involving the Enron email data set. Over 12,500 emails were classified by humans, into the Business and Personal categories. They used inter-annotator agreement to evaluate how well humans perform this task. They also used a probabilistic classifier based upon the distribution of distinguishing words, to determine the feasibility of separating business and personal emails by machine.

In [2] and [3], the authors trained their models on the Enron data set, and tested them on the Enron and Avocado data sets. In [2], the authors represented the email exchange networks as social networks with graph structures. They used social networks features from the graphs in addition to pre-trained GloVe embedding vectors as lexical features from email content to improve the performance of SVM and Extra-Trees classifiers. As a supplementary contribution to this paper, the authors also provided manually annotated sets of the Enron and Avocado email corpora. In [3], the same authors additionally considered the thread structure of emails which improved the performance further. They also used node embedding based on both lexical and social network information. All results presented here are used in Section 6 for comparison purposes.

There are a lot of other research papers that cover solving different email processing tasks, such as spam-filtering, multi folder classification, phishing email classification [26], but we have focused here only on the papers most related to our research.

3. Our Approach

The rich textual structure of email has a predefined format in which two main segments have been identified: a header and conversational content. Our approach exploits useful information from both of them. The textual features used in the classification process are based on the conversational content only, ignoring the content of email headers, e.g. dates, personal names, etc. Email domains identified with regular expressions from the headers are added to the end of the email content (the most recent email or the most recent email with quote messages from the same thread arc). We have found that the result of this is that the words such as 'hotmail' and 'yahoo' are characteristics of the Personal class (see Table 2). Personal communication often happens between people outside the organization and email domains could be an indicator of it. The architecture of our approach is presented in Fig. 1.

Based on the fact that emails from the same thread, and especially from the thread arc, usually belong to the same Business/Personal class [3], we have split our experiments based on the content used as follows:

- The most recent email (E – baseline)
- The most recent email with domains found in headers (ED)
- The most recent email with quote messages from the same thread arc (EQ)
- The most recent email with quote messages from the same thread arc and domains found in headers (EQD)
- A whole email thread arc found in the body field (B – baseline)

In all mentioned cases, the subject is added to the email content. To obtain some new/fresh insights and results, we have analysed user writing behavior in the business environ-

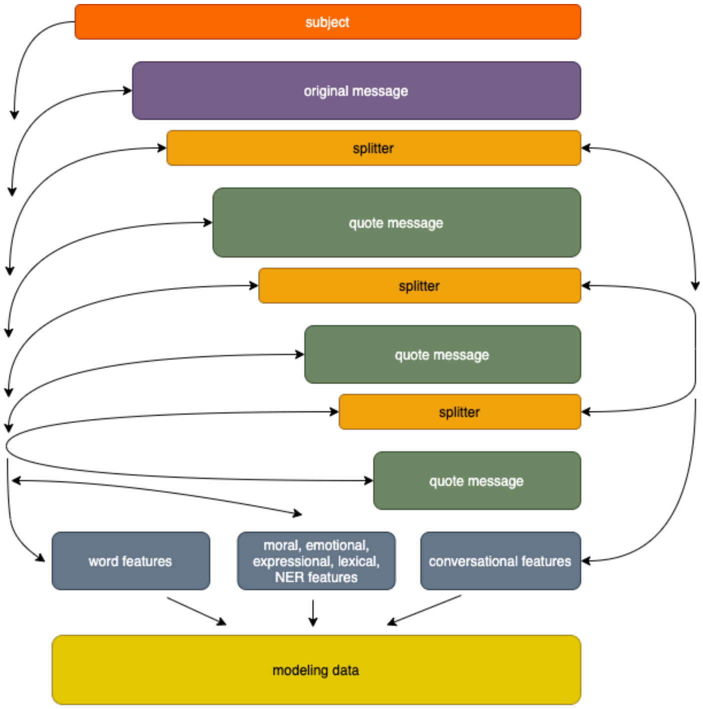

Fig. 1. The architecture of the proposed approach for effective emails classification into Business and Personal classes

ment and conversational context from a lexical, conversational, expressive, emotional, and moral perspectives.

3.1. Data

The Enron email data set consists of both personal and business emails from over a hundred Enron employees over a period for 3.5 years (1998 to 2002). It was made publicly available by the Federal Energy Regulatory Commission (FERC) during the legal investigation of the company's collapse. The corpus was first processed and released by Klimt and Yang (2004) at Carnegie Mellon University (CMU), and this CMU data set has later been re-processed by several other research groups. In our experiments, we use the version of annotated Enron data set presented in the article [2] which we call *Enron Columbia* and denote with $Enron_C$. This data set is annotated as follows:

- Business, with clearly professional content;
- Somehow Business, with professional content but with some personal parts;
- Mixed, with combined both professional and personal content;
- Somehow Personal, with personal content but with some business-related parts;
- Personal, with clearly personal content;
- Cannot Determine, with not enough content to determine the category.

In our experiments, these categories were merged into two categories: Business and Personal in the following way:

- Personal: Personal, Somehow Personal, Mixed;
- Buisness: Buisness and Somehow Business.

Fig. 2 presents $Enron_C$ data set through a frequency scatter plot [2] of the words present in the Business and Personal categories. The word frequency metric is what scattertext uses as the coordinates for each point. The x-axis indicates the frequency in the Business category: if a word frequently appears in business emails, it is placed on the right. Similarly, the y-axis encodes the frequency in the Personal category. A word that frequently appears in personal emails will be placed on the top area. Consequently, the areas where the more frequent words appear is of particular interest: top left (frequent in personal emails), bottom right (frequent in business emails), and top right of the figure (frequent in both personal and business emails). These areas offer a view of how the words are distributed in these two categories. For example, common Business words such as 'agreement', 'energy', and 'attachment', stress the official tone that can be found in the narratives of the business emails. In contrast, the emails found in the Personal category have a more relaxing tone, frequently using words such as 'love', 'weekend', and 'fun'. The colors express the value of the score called "scaled F-Score" introduced by the authors in [13]. Words with scores near zero, colored in yellow and orange in the plot, have frequencies that are similar for both classes. These words are not of great importance. When the frequency of the word is dominated by one class, scores are shifting to -1 (Business) or 1 (Personal), marked with red or blue color respectively. The darker the color of red or blue indicates the higher dominance of the word for the corresponding class. Another version of the Enron data set is annotated by Berkeley students in Applied Natural Language Processing Course (ANLP), so we call it *Enron Berkeley* and denote it with $Enron_B$. They developed a set of hierarchical categories and the selected subset of emails focusing on business-related emails. Each email message was annotated by two people and got assigned multiple labels at once. The data set contains 1702 emails that were categorized into 53 topic categories, such as company strategy, humor, and legal advice. It has been mainly used as a benchmark data set for multi-label classification. In our experiments these categories were merged as:

- Personal: Purely Personal, Personal in a professional context and Private humor;
- Buisness: all other categories which are related to business policies, strategy, legal notes or regulations.

The numbers of emails for each category in both Enron distributions (*Enron Columbia* and *Enron Berkeley*) are presented in the Table 1, while top business and top personal words in both data sets are presented in Table 2.

3.2. Preprocessing and Text Representation

The content of the emails is represented in the forms of vectors of word frequencies (or Bag of Words denoted as BoW in the following part of the text), Tf-Idf vectors on n-grams and n-gram characters as well as BERT embeddings. Elements of Tf-Idf matrix are calculated using the formula presented in equation 1:

[2] https://github.com/JasonKessler/scattertext

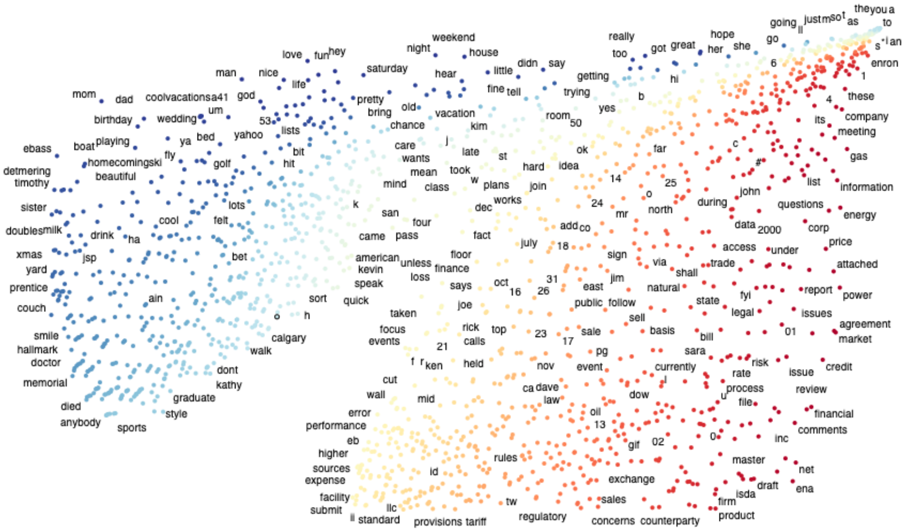

Fig. 2. Characteristic words for Business and Personal classes. 'Love', 'weekend', 'fun' in Personal and 'agreement', 'energy', 'attachment', words containing numbers in Business are among the most dominant words

Table 1. Summary of Enron data sets before and after processing empty and duplicate emails

Data Set	Business	Personal	Total
$Enron_C$	9738 (86.5%)	1523 (13.5%)	11261
$Enron_{Cp}$	8651 (86.6%)	1340 (13.4%)	9991
$Enron_B$	1491 (87.6%)	211 (12.4%)	1702
$Enron_{Bp}$	1276 (88.0%)	173 (11.9%)	1449

Table 2. Dominant words for Business and Personal class and characteristic words for the whole corpus

data set	Top Business	Top Personal	Corpus Characteristic
$Enron_C$	energy, agreement, information, power, market, attached, gas, price, trading, issues, credit, review, questions, contract	love, hotmail, night, weekend, hey, msn, man, mom, yahoo, fun, god, really, game, house	enron, ferc, skadden, hotmail, isda, http, attached, dynegy, aol, fyi, carrfut, counterparty, ect, cpuc, com, org, trading, nymex, eol, thanks, www, enrononline, gas, tomorrow, calpine, pge
$Enron_B$	state, gas, price, energy, market, electricity, utility, power, blackouts, billion, federal, percent	thanks, sorry, great, love, life, congratulations, london, studio	2001, blackouts, enron, dynegy, edison, generators, electricity, 2000, megawatt, deregulation, gov

$$w_{i,j} = tf_{i,j} * log(\frac{N}{df_i}) \tag{1}$$

where $w_{i,j}$ is Tf-Idf weight for token i in document j, tf_i is the number of occurrences of token i in document j, df_i is the number of documents that contain token j and N is the total number of emails in the training set. Vocabulary size for different BoW and Tf-Idf text representations and experiments is presented in the Table 3.

Lemmatization was included in the data preprocessing stage for verbs, nouns, adjectives, and adverbs. For lemmatization we used WordNetLemmatizer from NLTK [3] python package. We had special treatment of numbers, personal names, punctuations, spaces, and contractions. Also, we defined corpus-specific stop words. The author in [13] introduces measures of precision and recall for the words in the corpus and explains their inverse relationship. Precision is a word's discriminative power regardless of its frequency, while recall denotes the frequency at which a word appears in a particular class, or $P(word|class)$. For visual interpretation, the words with high recall values tend toward the top right-hand corner of the chart, while the words with high precision values tend toward the axes (See Fig. 2). The revelation that extremely high recall words tend to be stop words is used for the creation of the corpus specific list of stop words.

To compare different preprocessing and text representation techniques, we performed a large set of experiments using SGD-SVM ML algorithm. As it is presented in the Fig. 3.2, we came to conclusion that word 2-grams outperforms word n-grams of other lengths ($n = 1$ or $n > 2$). Moreover, Tf-Idf weights outperform frequency weights which were widely used in the previous publications on the same task. Additionally, word n-grams outperform character n-grams. Limiting minimum or maximum number (or percentage) of allowed token appearance across the corpus does not improve model performance by any means. Incorporation of lemmatization and custom defined stop words plays an important role in improving the model performance, together with the removal of personal names and punctuation tokens. However, limitation of vocabulary size decreases model performance. From all of these points, best resulting preprocessing actions have been applied on the raw text, resulting in the text representation used in the following experimentation steps.

Table 3. Vocabulary size for different text representations and experiments - E - the most recent email, ED - email with domains, EQ - email with quotes, EQD - email with quotes and domains, B - body used as email content

Experiment	BoW/Tf-Idf (1,1)	Tf-Idf-Ngram (1,2)	Tf-Idf-Ngram-Char (1,4)	BERT embeddings
E	22381	217186	100417	30522
ED	23624	223595	105819	30522
EQ	29074	320018	123615	30522
EQD	30257	325929	128274	30522
B	33099	362959	138409	30522

Another technique for emails representation used in our work is BERT embeddings vectors. Word embeddings techniques aim to use continuous low-dimension vectors rep-

[3] https://nltk.org/

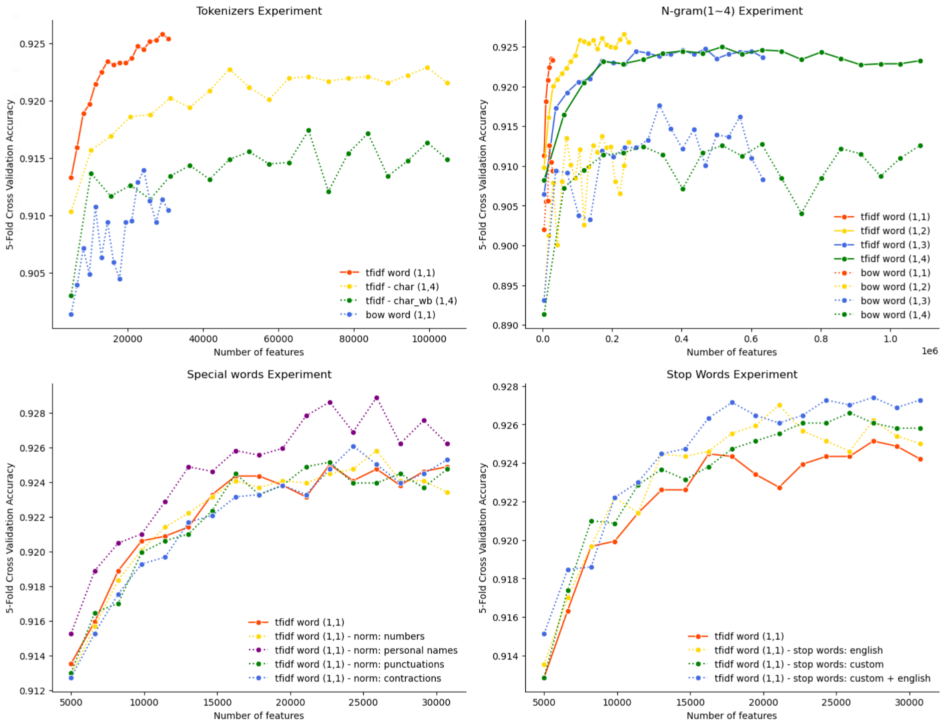

Fig. 3. The results of testing preprocessing steps for different weight and token types, n-gram lenght, stop words and special words selection. Used SGD classifier with default set of parameters in email with domains (ED) experiment

resenting the features of the words captured in context [15]. A compact pre-trained BERT word embedding model, pre-trained on Wikipedia and BookCorpus, was selected from Google's TensorFlow Hub [4] repository. This model, with L=2 hidden layers (i.e., transformer blocks), a hidden size of H=256, and A=4 attention heads, was used for initializing the word embedding layer (the input layer) of all our deep learning models. Model architecture and training objectives from the standard BERT model is replicated to a wide range of model sizes, making smaller BERT models applicable for environments with restricted computational resources. They can be fine-tuned in the same manner as the original BERT models [27]. Descriptive statistics of email content lengths for different input emails (E, ED, EQ, EQD, B) and Tukey's outliers rule, helped us to set appropriate thresholds for maximum sequence length. In the case of E and ED, it is set to 256, while in the case of EQ, EQD and B, it is set to 512 to gather most of the information from the data.

[4] https://tfhub.dev/s?module-type=text-embedding

3.3. Additional Features

Set of additional lexical (including punctuation-based and NER-based), conversational, expressional, emotional, and moral features has been extracted to analyse conversational context of exchanged emails.

Lexical Features (Lex) capture various counts and ratios associated with the subject and content of the email. Text classification extensively relies on such features and hence we hypothesize that the lexical properties will contribute to our task. Syntactic features include NER–based features, number of lines, number of noun phrases, syllables, difficult words which contain more than one syllable, average syllable per word (ASPW) and sentence (ASPS), sentence and word density. ASPW, sentence and word densities are defined with equations 2, 3 and 4 respectively:

$$ASPW = \frac{\#syllables}{\#words} \tag{2}$$

$$sentence\ density = \frac{\#sentences}{1 + \#lines} \tag{3}$$

$$words\ density = \frac{\#words}{1 + \#spaces} \tag{4}$$

where #sentences, #words, #lines and #spaces denote number of, sentences, words, lines (including blank lines) and blank spaces in email content respectively. To identify syllables, *pyphen* python package is used, while remaining features in this group are calculated with *textstat* package.

The business indicator is a numerical feature representing the ratio of business terms in the content. Business terms are identified using Business Thesaurus [5] dictionary containing terms, expressions, and terminology used in business conversations. The ratio of abbreviations in the content is noted as an acronyms indicator. Abbreviations are identified using Abbreviations and Acronyms Dictionary [6] together with regular expressions to fine tune their finding in the email content.

Punctuation-based features (Punct) measure the presence of dots, question marks, exclamation marks, hash and reference tags with their ratio among the whole punctuation characters found in the email content.

NER-based features (NER) are numerical representations of the NER tags presence in the content. The ratio of personal names, organization names, words containing numbers, words in English language marked as connectors (e.g. ’in’, ’the’, ’all’, ’for’, ’and’, ’on’, ’but’, ’at’, ’of’, ’to’, ’a’), month and day names, and a valid email and URL addresses are incorporated in the list of features.

Conversational Features (Conv) are extracted from email signatures containing the number of mail recipients and information about a conversation with external email domains. For that purpose we use free email domains dictionary [7]. The free domains ratio is the ratio of free domains presented in the email content including signature among all

[5] https://www.businessballs.com/glossaries-and-terminology/business-thesaurus-290/

[6] https://abbreviations.yourdictionary.com/

[7] https://github.com/willwhite/freemail

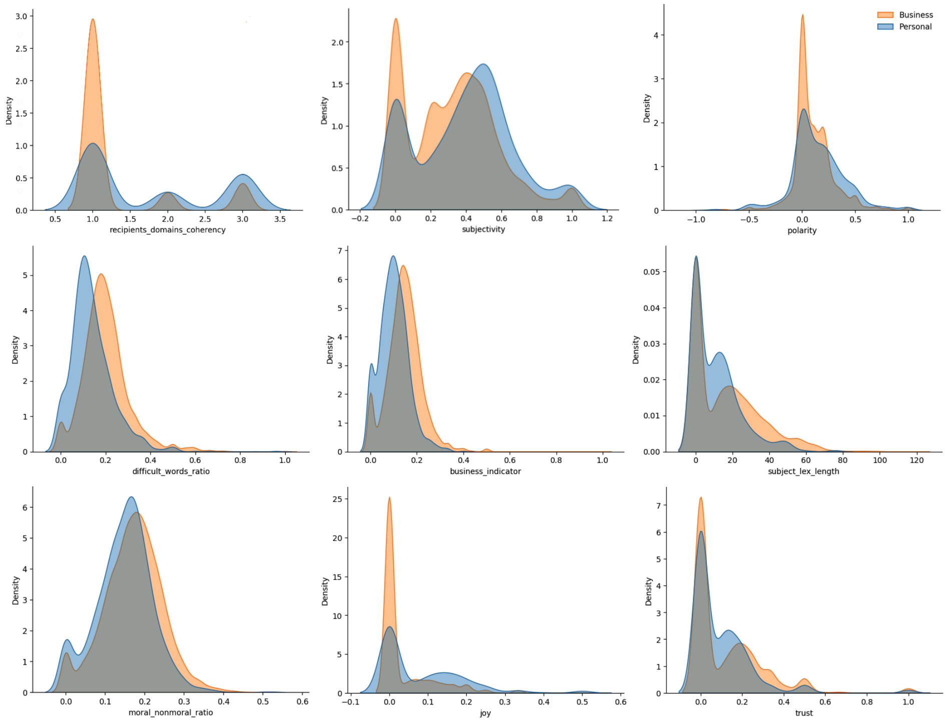

Fig. 4. Variation in the features distributions in the Business and Personal classes

domains found there. Recipients domains coherency is the feature created to capture coherency between recipients domains e.g. if they all belong to the default company domain, external domains or if recipient domains are of a mixed structure.

Expressional Features (Expr) capture information such as readability of text, subjectivity and polarity. Subjectivity and polarity are based on the TextBlob [8] implementation. Polarity is a float that lies in the range of [-1,1] where 1 denotes a positive statement and -1 denotes a negative statement. Subjective sentences refer to personal opinion, emotion or judgment, whereas objective ones refer to factual information. Subjectivity is presented as a float number that lies in the range of [0,1] closer to 1 in a more subjective context. Readability is measured based on Automated readability index (ARI) and Flech Reading Ease Score (FRES), which are calculated by equations 5 and 6:

$$ARI = 4.71(\frac{\#characters}{\#words}) + 0.5(\frac{\#words}{\#sentences}) - 21.43 \tag{5}$$

$$FRES = 206.835 - 1.015(\frac{\#words}{\#sentences}) - 84.6(\frac{\#syllables}{\#words}) \tag{6}$$

where #characters, #words , #sentences and #syllables denote the number of letters and numbers, words, sentences and syllables in the text.

[8] https://textblob.readthedocs.io/en/dev/

The LWM Algorithm is giving a 'grade level' measure, reflecting the estimated years of education needed for reading the text fluently. ARI and FRES scores measure how easy it is to read a text. We use textstat [9] python package for the implementation of that.

Table 4. Summary of features. Meta refers to all extracted features combined together

Feature Group		Features List	# of Features
Lexical		Number of characters and words in content and subject, sentences count, average sentence length, average word length, noun phrases, average syllables per word, average syllables per sentence, sentence and word density, difficult words, business indicator, acronyms indicator	16
	NER-based	Ratio of personal name, organization name, number, connectors, month name, day name, email and url address tags	8
	Punctuation-based	Ratio of dots, question marks, exclamation marks, hash tags, reference tags	5
Conversational		Free domains in headers ratio, number of recipients, recipients domains coherency	3
Expressional		Automated Readability Index(ARI), Flech Reading Ease Score(FRES), Linsear Write Metric(LWM), content subjectivity, content polarity	5
Moral		Probability measures of care, sanctity, authority, loyalty and fairness on word and sentence, moral/non-moral ratio	11
Emotional		Measures of trust, joy, anger, disgust, sadness, fear, surprise, positive, negative	9
All features (Meta)			57

Emotional Features (Emo) use the Plutchik's approach [19] which postulates the following eight basic human emotions: joy, sadness, anger, fear, trust, disgust, anticipation, and surprise, extending a simple positive-negative dichotomy to capture the full range of emotions. There have been extensive applications of this approach, for example, the National Research Council (NRC) Word-Emotion Association Lexicon which contains 10,170 lexical items that are coded for Plutchik's basic human emotions [16]. Plutchik's categories also have the advantage of providing a balanced list of positive (trust, joy, anger, and anticipation) and negative (disgust, sadness, fear, and surprise) emotions. To the best of our knowledge, they have not been applied in business conversation analysis in general, or email content analysis in particular. We use the python NRCLex [10] package which expands the lexicon to 27,000 words based on WordNet synonyms and effectively measures the emotional effect on the categories.

Moral Features (Mor) are based on Moral Foundations Theory (MFT), a framework for explaining variation in people's moral reasoning [6]. The framework decomposes the types of moral evaluations people make into five foundations: Authority/Subversion, Care/Harm, Fairness/Cheating, Loyalty/Betrayal, Sanctity/Purity. The emphasis on moral

[9] https://pypi.org/project/textstat/0.1.6/

[10] https://pypi.org/project/NRCLex/

foundations is most commonly inferred from written text (speech acts) by flagging combinations of words that have validated connections to each foundation. Recent behavioral research has focused on developing extended vocabulary sets with human ratings for mapping large sets of terms onto various moral foundations [11].

In this paper, we use an eMFD [11] python package to calculate the moral sentiment in emails. Each email document is assigned to five foundation probabilities that denote the average probabilty of each document belonging to one of the five moral foundations and five sentiment scores that describe the average sentiment of detected moral words for that foundation. In addition, the moral-to-nonmoral word ratio has been added to the list of moral features for that document.

All groups of features are merged into a single list, denoted with Meta, which contains 57 features in total (see Table 4). L2 normalization technique is applied to all constructed features, rescaling the features vector representation of each document to have Euclidean norm equals to 1. The graphs on Fig. 4 show how specific features vary in their distributions for the Business and Personal classes. Subject length, difficult and moral words ratio tend to have higher values for Business class. The Personal class has higher subjectivity and polarity scores, as well as words assigned to joy. Recipients domain coherency has higher values in the Personal class for the recipients domains outside the default organization domain (value 3).

When features are collinear, dropping one feature will have little effect on the model's performance because it can get the same information from a correlated feature. Multicollinear and correlated features from our set of Meta features were removed by performing hierarchical clustering on the Spearman rank-order correlations, with a threshold of 0.7, and a single feature from each cluster is kept. On the retrieved 'clean list' of features, we performed drop-column algorithm to measure the impact of each feature on the variance of the default model accuracy and use this measure as the driver for feature selection. With a threshold of 0.01, the final set of uncorrelated and the most important Meta features for each experiment was selected. We can observe that features from each predefined group of features take their role among the most important features on average, but also in each of the examined experiments. The most important features from each group are: joy, fear and trust (Emo); fairness, authority and loyalty (Mor); names, numbers and connector ratios (NER); exclamation and dots ratio (Punct); recipient domains coherency and free domains ratio (Con); subjectivity, polarity, FRES, LWF and ARI (Expr); business and acronyms indicators, average syllables per word, average word length, subject length, sentence and word density (Lex). Moreover, Expr, Mor and Emo features has the highest tendency towards the top. It is noticable that positive/negative (NRCLex) and polarity (TextBlob) features which are extracted using different packages and their lexicons have different scores in B experiment, which is not expected. It could be due to the differences in lexicons for these categories as well as the characteristics of particular content experiment. Moreover, in B experiment, some of the most important features such as names and connector ratios, FRES, difficult words, and free domains ratios indicate that header content which is included in the email body has an important role in distinguishing Business/Personal classes. Personal names, dates, and domain names (especially corporate domain enron.com), together with connectors and punctuation characters which can be found in reply and forward email headers are data set dependant. We have tried to avoid

[11] https://github.com/medianeuroscience/emfd

this dependency by using another email content structure through our experiments with an ambition to improve classification generalization on other data sets.

3.4. Machine Learning Techniques

Traditional Learning. The learning process in ML is producing the function f by processing the samples of the training set (E). The function f maps email content E_n to one of the classes $C_k, k = 2$. The email content for each E_k is represented with the numeric features vectors. Therefore, as it is described with equation 7, feature vector extractor ϕ computes vectors of features for each email from E:

$$\phi(E_k) = (\phi_1(E_k), \ldots \phi_d(E_k)), \phi(E) \in R^d \tag{7}$$

representing a point in the d dimensional feature space. Moreover, the parameter vector that specifies the contributions of feature vectors to the prediction output is given with equation 8:

$$P = P_1, \ldots P_d, P \in R^d \tag{8}$$

Consequently, in equation 9, we mathematically express f by assembling both $\phi(E)$ and P:

$$f = \phi(E) * P \tag{9}$$

The Gradient Descent optimization algorithm aims to find the coefficient of f with a condition that minimizes the cost of the inaccuracy of predictions. It uses different coefficient values and the cost function estimates their values through the predicted results for each sample of the training set. The aforementioned process occurs by comparing the prediction result with the actual value to choose the lowest loss. The algorithm tries different coefficient values to look for lower loss. The learning rate is used to update the coefficients for the next iteration. Such calculation is very expensive as the cost is computed over the entire training data set in each iteration. On the other hand, Stochastic Gradient Descent updates the coefficient for each training sample instead at the end of the iteration over all samples of the training set. We will apply the Stochastic Gradient Descent optimization technique in section 4 on a diverse set of linear classifiers and choose the best one for our task.

Extremely Randomized Trees (ERT) is an algorithm for building decision tree ensembles, for both supervised classification and regression problems. The best splitting attribute is selected for each node from a random subset of attributes. Including randomness in the cut-point choice, the algorithm builds an ensemble of decision trees whose structure is independent of the output values [5].

A comparison of the SGD and ERT classifiers performances was made by using their implementations from scikit-learn python library [18].

Deep Learning. With a successful initial application to computer vision problems, Convolutional Neural Networks (CNNs) confirmed their good performance in NLP [29]. CNNs are able to extract the local n-gram features, having difficulty with capturing long-distance dependencies.

Recurrent Neural Networks (RNNs) can capture dynamic information in serial data by recurrently connecting the hidden layer nodes. RNNs can store a state of context, learn and express relevant information in any long context window, unlike CNN's fixed-input formation. An RNN can overcome the problem of a long-distance dependency. However, it is difficult to train because gradients may explode or vanish over long sequences [10].

One way to address this problem is by employing a variant of the regular RNN, the LSTM [9]. LSTMs have a more complex internal structure with cells replacing RNN nodes, which allows LSTMs to remember information for either a long or short time. A regular LSTM tends to ignore future contextual information while processing sequences.

The Bidirectional LSTM (BiLSTM) is able to use both past and future contexts by processing the text from both directions [24].

Employing an attention mechanism between sequences (BiLSTM+Att), BiLSTM shows a considerable improvement by changing the contribution of each word to the analysis of the whole text [23], [22]. Before the RNN model summarizes the hidden states for the output, an attention mechanism amplifies the results by aggregating the hidden states (See equations 10 and 11) and weighting their relative importance (See equation 12), where W_h and b_h are the weight and bias from the attention layer.

$$e_i = tanh(W_h h_i + b_h), e_i \in [-1, 1] \tag{10}$$

$$w_i = \frac{exp(e_i)}{\sum_{t=1}^{N} exp(e_t)}, \sum_{i=1}^{N} w_i = 1 \tag{11}$$

$$r = \sum_{i=1}^{N} w_i h_i, r \in R^{2L} \tag{12}$$

Not all words make the same contribution to the business vs. personal categorization of the text. The attention mechanism is able to shuffle the word annotation weights according to their importance to the meaning of a sentence.

4. Experimental Framework

We compare the accuracy of the two traditional (SGD and ERT) and two deep learning (BiLSTM, BiLSTM+Att) algorithms on different vector representations of email content. The best parameters for both classical learners were selected by grid search algorithm as it is presented in Table 5.

Selected modified huber loss is equivalent to quadratically smoothed SVM with $gamma = 2$ [28]. In the following part of the text, SGD-SVM will denote modified huber loss. For deep learning models, parameters were selected manually using extensive experimentation. For all models, we use the binary cross-entropy loss function and the same optimizer that BERT was originally trained with: the 'Adaptive Moments' (Adam). This optimizer minimizes the prediction loss and does regularization by the weight decay, which is also known as AdamW. For the learning rate, we use the same schedule as BERT pre-training: the linear decay of a notional initial learning rate, prefixed with a linear warm-up phase over the first 10% of the training steps known as the number of the warm-up steps. The learning rate is set on 3e-5, being in line with the BERT paper [4], which specifies the initial learning rate values for fine-tuning. An early stopping strategy is used to prevent over-fitting [20]. All models use gradient descent with mini-batches of

Table 5. Grid-search parameter selection. B: Business, P: Personal. Balanced: class weights are adjusted inversely proportional to class frequencies in the training set

Classifier	Parameter	Search Parameter Space	Best Performing Values
SGD	loss	hinge, log, modified_huber, squared_hinge, perceptron	modified_huber
	penalty	l1, l2, elasticnet	l2
	alpha	0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000	0.0001
	learning_rate	constant, optimal, invscaling, adaptive	optimal
	class_weight	{P: 0.5, B: 0.5}, {P: 0.6, B: 0.4}, {P: 0.7, B: 0.3}, balanced	{P: 0.7, B: 0.3}
	eta0	1, 10, 100	10
ERT	n_estimators	10, 20, 30, 50, 100, 200	10
	criterion	gini, entropy	entropy
	min_samples_leaf	1, 3, 10	3
	class_weight	{P:0.5, B:0.5}, {P:0.6, B:0.4}, {P:0.7, B:0.3}, balanced	{P:0.7, B:0.3}

size 64, ReLU activation function, dimension of word embedding equal to 256, maximum sequence length equal to 256 (E, ED) and 512 (EQ, EQD, B), the number of LSTM equal to maximum sequence length, dropout ratio of 0.1 for LSTM, and 0.4 for Dense layers for all models. Models are trained on 30 epochs.

Models are trained on the training set and evaluate the prediction with the best scores retrieved on the validation and test sets. For traditional models (SGD-SVM, ERT), we use cross validation with 5 folds. For deep learning models, the split ratio for training, validation, and test sets is 50:25:25. In order to illustrate the good performance of our approach, we compare the results with baseline models built on the most recent email (E) and the whole thread arc from the body field of the data set (B).

For evaluating the performance of the techniques, we use the typical evaluation metrics that come from information retrieval - precision, recall and F1 measure, accuracy and balanced accuracy. We aim to improve both the general and balanced accuracy of the classification model as well as F1 measure on minority Personal class.

5. Experimental Results

The results of the model comparison of BoW, Tf-Idf and BERT word embedding with and without Meta features included for SGD-SVM and ERT classifiers in ED experiment are presented in Table 6. Our results show that using Tf-Idf weights for unigrams, $n = 1$ (Tf-Idf-Unigram), unigrams and bigrams, $n \in [1, 2]$ (Tf-Idf-Ngram) and ngram characters of length 1-4 (Tf-Idf-Ngram-Char) as features significantly improves model performances compared to BoW weights on unigrams, $n = 1$ used as features. Moreover, Tf-Idf-Ngram weights generally give the best performance across the experiments and measures.

Traditional learners on all Tf-Idf weights have comparable metric values with deep learning learners and even overcome them at the learner general accuracy, while the later give better balanced accuracy and F1 score on minority Personal class. ERT classifier presents lower values across the measures compared with SGD-SVM classifier. From the results shown, we can also observe that the BiLSTM+Att obtains higher scores than the BiLSTM without the attention mechanism. Moreover, all models with additional Meta features are showing better results improving it by at least 0.1% across the experiments.

Table 6. Comparison between traditional (SGD-SVM, ERT) and deep learning algorithms (BiLSTM, BiLSTM+Att) for different email content representations with and without additional email features included for emails content with domains experiment (ED)

				Business			Personal		
Algorithm	Features	Accuracy	Balanced Accuracy	Precision	Recall	F1	Precision	Recall	F1
ERT	BoW	90.1%	68.9%	91.4%	97.8%	94.5%	73.9%	39.9%	51.9%
	BoW + Meta	90.7%	70.5%	91.8%	98.0%	94.8%	76.9%	42.9%	55.1%
	Tf-Idf-Unigram	90.2%	69.5%	91.5%	97.8%	94.6%	74.1%	41.1%	52.9%
	Tf-Idf-Unigram + Meta	90.6%	70.5%	91.8%	97.8%	94.7%	75.4%	43.2%	55.0%
	Tf-Idf-Ngram	90.1%	67.2%	90.9%	98.4%	94.5%	77.4%	36.0%	49.2%
	Tf-Idf-Ngram + Meta	90.3%	68.9%	91.4%	98.1%	94.6%	76.3%	39.6%	52.2%
	Tf-Idf-Ngram-Chr	89.8%	66.2%	90.6%	98.4%	94.4%	76.9%	33.9%	47.1%
	Tf-Idf-Ngram-Chr + Meta	90.1%	67.6%	91.0%	98.3%	94.5%	76.9%	36.9%	49.9%
SGD-SVM	BoW	90.0%	79.9%	94.7%	93.7%	94.2%	62.0%	66.0%	64.0%
	BoW + Meta	90.2%	80.2%	94.7%	94.0%	94.3%	63.0%	66.4%	64.6%
	Tf-Idf-Unigram	92.0%	82.6%	95.3%	95.5%	95,4%	70.7%	69.8%	70.2%
	Tf-Idf-Unigram + Meta	92.5%	82.1%	95.1%	96.4%	95.7%	74.4%	67.9%	71.0%
	Tf-Idf-Ngram	**92.8%**	83.2%	95.4%	96.4%	**95.9%**	75.0%	70.1%	72.5%
	Tf-Idf-Ngram + Meta	**92.9%**	**83.8%**	95.6%	96.3%	**95.9%**	74.8%	71.3%	**73.0%**
	Tf-Idf-Ngram-Chr	92.3%	82.3%	95.2%	96.0%	95.6%	72.6%	68.5%	70.5%
	Tf-Idf-Ngram-Chr + Meta	92.2%	83.0%	95.4%	95.6%	95.5%	71.1%	70.4%	70.7%
BiLSTM	BERT-Embd	91.4%	81.1%	94.5%	95.6%	95.0%	71.5%	66.7%	69.0%
	BERT-Embd + Meta	91.5%	81.8%	95.3%	94.9%	95.1%	67.0%	68.6%	67.8%
BiLSTM+Att	BERT-Embd	92.1%	83.4%	95.9%	95.1%	95.5%	67.6%	71.7%	69.6%
	BERT-Embd + Meta	92.3%	83.4%	95.7%	95.4%	95.5%	70.3%	71.3%	70.8%

The best models from traditional and deep learning streams, SGD-SVM and BiLSTM+Att on Tf-Idf-Ngram + Meta and BERT-Embd +Meta vector spaces from the previous results have been selected for comparison of the email content experiments. The ED and EQD experiments were able to capture additional knowledge of each email content, so that the whole system slightly improved accuracy compared with the E and EQ experiments respectively, but for such high accuracy values, any improvement becomes significant. The EQD experiment made full use of the associated data available in each email (quotes and recipient email domains) with retrieved improvement in Accuracy score for 3.2% and for 0.6% in SGD-SVM classifier compared with the baseline E and B experiments respectively, as it is presented in Table 7.

Testing approach generalization has been performed using the models built on $Enron_{Cp}$ and $Enron_{Bp}$ data sets independently. For the model trained on $Enron_{Cp}$, the whole $Enron_{Bp}$ data set has been used for testing. In the $Enron_{Bp}$ based model, a data set is firstly split on training, validation, and test data sets. The results from this test, on all different text representations on SGD-SVM and BiLSTM classifiers, confirm that a model can capture important information and transfer the knowledge to differently annotated data sets (see Table 8). Even more, the ED experiment better generalizes the learning process than the B experiment, with a lower difference on all measures between the models. We observe a slight decrease in the test results on $Enron_{Bp}$ since our model parameters are optimized on the $Enron_{Cp}$ data set. Moreover, the size of $Enron_{Bp}$ is much smaller, it is not intentionally annotated for business/personal categorization and it

Table 7. Comparison between different email content representations (Experiments - E, ED, EQ, EQD, B) with additional email features included. SGD-SVM and Bi-LSTM+Attention algorithms are used for models building and testing

					Business			Personal		
Experiment	Algorithm	Features	Accuracy	Balanced Accuracy	Precision	Recall	F1	Precision	Recall	F1
E	SGD-SVM	Tf-Idf-Ngram + Meta	**92.6%**	81.7%	94.9%	96.7%	**95.8%**	**75.6%**	66.7%	**70.9%**
	BiLSTM+Att	BERT-Embd + Meta	92.3%	**86.0%**	97.5%	93.8%	95.6%	58.3%	78.2%	66.8%
ED	SGD-SVM	Tf-Idf-Ngram + Meta	**92.9%**	**83.8%**	95.6%	96.3%	**95.9%**	74.8%	71.3%	**73.0%**
	BiLSTM+Att	BERT-Embd + Meta	92.3%	83.4%	95.7%	95.4%	95.5%	70.3%	71.3%	70.8%
EQ	SGD-SVM	Tf-Idf-Ngram + Meta	**95.7%**	**90.3%**	97.4%	97.7%	**97.5%**	85.0%	82.9%	**83.9%**
	BiLSTM+Att	BERT-Embd + Meta	94.1%	87.2%	96.6%	96.6%	96.6%	77.8%	77.8%	77.8%
EQD	SGD-SVM	Tf-Idf-Ngram + Meta	**95.8%**	**91.3%**	97.7%	97.5%	**97.6%**	84.0%	85.0%	**84.5%**
	BiLSTM+Att	BERT-Embd + Meta	94.0%	87.7%	97.0%	96.0%	96.5%	73.9%	79.4%	76.5%
B	SGD-SVM	Tf-Idf-Ngram + Meta	**95.2%**	**90.2%**	97.4%	97.1%	**97.3%**	81.9%	83.2%	**82.5%**
	BiLSTM+Att	BERT-Embd + Meta	93.9%	86.0%	95.5%	97.4%	96.4%	83.8%	74.5%	78.9%

contains initial categories such as ’personal in professional context’ included in the final Personal class email categorization.

6. Comparison with Other SOA Methods

To the best of our knowledge, there have been three attempts in research papers published so far to classify emails in Business and Personal categories. All of them have used their own annotated emails of the Enron corpus with different classification strategies and compared obtained results with other available annotated email data sets (Enron, Avocado). Since our work is based only on the Enron data set, we will compare the retrieved results with the same and differently annotated Enron data sets. The results presented in the paper [12] are based on the Enron data set annotated by the authors, usually denoted as the Sheffield Enron data set in the research papers. It is not obvious, as it is also noted by [3], which training/test ratio was used for obtaining these results. Moreover, the structure of the email content used for email annotation and classification is not known to us. For that reason, we can only treat results from [12] as general points for our classification results comparison. The results obtained after the application of the models from our approach outperform the reported results in the overall Accuracy, Recall, and F1 score on minority (Personal) class in the EQ, EQD and B experiments.

On the other hand, the annotated data set presented in [2] was used in our work. When compared, the results obtained in our baseline experiment E outperform the results reported in the papers [2] and [3] in the overall Accuracy score (+1.4/+1.6%). Macro F1 score on minority Personal class in the E and ED experiments (+0.4% and +2.5% respectively) is better than the one presented in [3]. By comparing other measures from the classification report, they outperform results reported in both of these papers in overall Accuracy score (+4.6%), Recall (+4.9%) and Macro F1 (+2.9%) on Business and Macro F1 (+6.4%) on Personal class across the EQ, EQD and B experiments (see Table 9). Although it is not noted if the authors treated only the most recent email or the whole

Table 8. Results of testing the models for emails content with domains (ED) and body (B) experiments on Berkeley data set - $Enron_{Bp}$

			Exp=ED				Exp=B			
Train Data	Algorithm	Features	Accuracy	Balanced Accuracy	F1 Macro	F1 Weighted	Accuracy	Balanced Accuracy	F1 Macro	F1 Weighted
$Enron_{Cp}$	SGD-SVM	BoW	83.7%	60.0%	60.3%	83.5%	87.1%	54.4%	55.2%	83.8%
		BoW + Meta	85.9%	57.0%	58.3%	84.1%	87.8%	56.5%	58.3%	84.9%
		Tf-Idf-Unigram	87.9%	59.2%	61.6%	85.8%	**88.5%**	57.8%	60.2%	85.8%
		Tf-Idf-Unigram + Meta	87.3%	58.1%	60.0%	85.2%	**88.5%**	58.8%	**61.4%**	86.0%
		Tf-Idf-Ngram	88.0%	58.0%	60.2%	85.5%	88.4%	54.7%	55.7%	84.7%
		Tf-Idf-Ngram + Meta	89.0%	58.0%	60.6%	86.1%	88.4%	56.4%	58.3%	85.3%
		Tf-Idf-Ngram-Chr	87.2%	59.3%	61.3%	85.4%	88.1%	54.5%	55.4%	84.5%
		Tf-Idf-Ngram-Chr + Meta	88.3%	59.1%	61.7%	86.0%	87.7%	54.1%	54.7%	84.1%
	BiLSTM	BERT-Embd	87.8%	69.2%	61.3%	89.1%	87.3%	65.6%	57.1%	90.1%
		BERT-Embd + Meta	87.9%	69.4%	61.6%	90.1%	87.9%	69.4%	54.6%	91.7%
	BiLSTM+Att	BERT-Embd	88.7%	73.2%	**64.1%**	90.6%	87.5%	66.6%	56.5%	90.7%
		BERT-Embd + Meta	**89.2%**	**78.4%**	62.3%	**91.9%**	88.1%	**70.9%**	58.8%	**91.0%**
$Enron_{Bp}$	SGD-SVM	BoW	86.5%	70.9%	67.4%	87.5%	87.5%	67.1%	69.1%	86.6%
		BoW + Meta	89.0%	69.7%	**69.2%**	89.1%	87.7%	69.5%	71.0%	87.2%
		Tf-Idf	88.7%	61.8%	63.3%	88.0%	90.6%	70.6%	71.7%	90.4%
		Tf-Idf + Meta	89.8%	63.7%	65.9%	88.9%	90.9%	69.4%	71.3%	90.5%
		Tf-Idf-Ngram	90.4%	58.9%	61.7%	88.4%	90.4%	65.3%	67.7%	89.5%
		Tf-Idf-Ngram + Meta	90.4%	60.2%	63.1%	88.7%	90.9%	68.2%	70.5%	90.3%
		Tf-Idf-Ngram-Chr	89.3%	59.6%	61.6%	87.9%	90.9%	73.3%	**73.6%**	90.9%
		Tf-Idf-Ngram-Chr + Meta	90.1%	61.3%	64.0%	88.7%	**92.0%**	67.5%	71.5%	91.0%
	BiLSTM	BERT-Embd	89.7%	55.4%	50.4%	93.0%	89.4%	72.5%	55.6%	92.3%
		BERT-Embd + Meta	90.7%	**71.1%**	61.0%	92.7%	89.7%	69.5%	52.4%	92.6%
	BiLSTM+Att	BERT-Embd	90.3%	45.3%	47.5%	**94.6%**	89.7%	74.5%	52.5%	93.0%
		BERT-Embd + Meta	**90.7%**	70.8%	56.6%	93.5%	90.4%	**88.4%**	56.9%	**93.9%**

thread arc stored in the Body field of $Enron_C$ data set as individual email, the latest observation has confirmed the strength of our approach in both of these cases.

7. Conclusion and Future Work

The importance and usage of emails by both personal and business users are continuously growing despite the prevalence of alternative means, such as instant mobile and social network messaging. Therefore, email management is an important and growing problem for individuals and organizations. In this paper, we have explored the classification of emails into two main categories, business and personal.

During our work, a comprehensive set of experiments was conducted to find the best solution or this task. We used different traditional and deep learning ML techniques including SGD-SVM, ERT, BiLSTM, and BiLSTM+Att together with different text representation techniques such as BoW and Tf-Idf, word and character n-grams, as well as BERT embeddings. The experimental results showed that traditional ML techniques with Tf-Idf text representation techniques slightly outperformed deep learning approach on this task. The reason for that may be the limitations in the research computational environment we used. Additionally, we put a lot of effort into introducing and experimenting with various additional features. To achieve the best possible generalization of the model, we excluded from the email all specificity of the training data set and focused only on the part of the email that contains the conversation itself.

Based on this work, we plan to expand our research in several different directions. First, data sets that differ in many aspects should be incorporated, including email data

Table 9. Comparison of results with other SOA methods. Accuracy, F1, Precision, and Recall measures were taken from the best experiments on test sets reported in the papers

			Business			Personal		
Paper		Accuracy	Precision	Recall	F1	Precision	Recall	F1
[12]		93.0%	92.0%	99.0%	95.0%	95.0%	69.0%	80.0%
[2]		91.2%	96.7%	92.1%	94.4%	73.5%	87.5%	79.9%
[3]		91.0%	96.6%	92.9%	94.7%	63.4%	79.3%	70.5%
Our Approach	E	92.6%	94.9%	96.7%	95.8%	75.6%	66.7%	70.9%
	ED	92.9%	95.6%	96.3%	95.9%	74.8%	71.3%	73.0%
	EQ	95.7%	97.4%	97.7%	97.5%	85.0%	82.9%	83.9%
	EQD	**95.8%**	97.7%	96.1%	**97.6%**	84.0%	85.0%	**84.5%**
	B	95.2%	97.4%	97.1%	97.3%	81.9%	83.2%	82.5%

sets in languages other than English and other conversational data sets, such as short messages. Further, different weighting schemes for additional features used in our research (such as NER tokens) should be investigated. Some of the lexicons, such as acronyms, business words and personal names, should be further analysed and improved. By using pre-trained BERT models based on the conversational data sets from business environments, as well as the sentence instead of the word embedding space for text representation could give significant value. Also, we plan to extend the research on the prediction of the hierarchical organizational structure by analyzing only business emails exchanged through the organization. One of our goals will be to examine extraction of the signatures from the business emails, as well as entities from their signatures. Our approach raises questions about the significance of different ways of expression in email communication, and how they can be used to better understand human behavior in a business environment. By understanding more deeply the emotional and moral framework of correspondents, organizers can better anticipate their response to certain requests and predict the outcome of the planned activities.

Acknowledgments. The work presented has been supported by the Ministry of Science and Technological Development, Republic of Serbia, through Projects No. 174021 and No. III47003.

References

1. Alhogail, A., Alsabih, A.: Applying machine learning and natural language processing to detect phishing email. Computers & Security 110, 102414 (2021)
2. Alkhereyf, S., Rambow, O.: Work hard, play hard: Email classification on the avocado and enron corpora. In: Proceedings of TextGraphs-11: the Workshop on Graph-based Methods for Natural Language Processing. pp. 57–65 (2017)
3. Alkhereyf, S., Rambow, O.: Email classification incorporating social networks and thread structure. In: Proceedings of The 12th Language Resources and Evaluation Conference. pp. 1336–1345 (2020)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
5. Geurts, P., Ernst, D., Wehenkel, L.: Extremely randomized trees. Machine learning 63(1), 3–42 (2006)

6. Graham, J., Haidt, J., Koleva, S., Motyl, M., Iyer, R., Wojcik, S.P., Ditto, P.H.: Moral foundations theory: The pragmatic validity of moral pluralism. In: Advances in experimental social psychology, vol. 47, pp. 55–130. Elsevier (2013)
7. Graovac, J.: A variant of n-gram based language-independent text categorization. Intelligent Data Analysis 18(4), 677–695 (2014)
8. Graovac, J., Kovačević, J., Pavlović-Lažetić, G.: Hierarchical vs. flat n-gram-based text categorization: can we do better? Computer Science and Information Systems 14(1), 103–121 (2017)
9. Graves, A.: Long short-term memory. In: Supervised sequence labelling with recurrent neural networks, pp. 37–45. Springer (2012)
10. Hochreiter, S.: The vanishing gradient problem during learning recurrent neural nets and problem solutions. International JOURNAL of Uncertainty, Fuzziness and Knowledge-Based Systems 6(02), 107–116 (1998)
11. Hopp, F.R., Fisher, J.T., Cornell, D., Huskey, R., Weber, R.: The extended moral foundations dictionary (emfd): Development and applications of a crowd-sourced approach to extracting moral intuitions from text. Behavior Research Methods 53(1), 232–246 (2021)
12. Jabbari, S., Allison, B., Guthrie, D., Guthrie, L.: Towards the orwellian nightmare: separation of business and personal emails. In: Proceedings of the COLING/ACL 2006 Main conference poster sessions. pp. 407–411 (2006)
13. Kessler, J.S.: Scattertext: a browser-based tool for visualizing how corpora differ. arXiv preprint arXiv:1703.00565 (2017)
14. Klimt, B., Yang, Y.: The enron corpus: A new dataset for email classification research. In: European Conference on Machine Learning. pp. 217–226. Springer (2004)
15. Mikolov, T., Yih, W.t., Zweig, G.: Linguistic regularities in continuous space word representations. In: Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies. pp. 746–751 (2013)
16. Mohammad, S.M.: Word affect intensities. arXiv preprint arXiv:1704.08798 (2017)
17. Nisar, N., Rakesh, N., Chhabra, M.: Review on email spam filtering techniques. International JOURNAL of Performability Engineering 17(2) (2021)
18. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al.: Scikit-learn: Machine learning in python. the JOURNAL of machine Learning research 12, 2825–2830 (2011)
19. Plutchik, R.: The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. American scientist 89(4), 344–350 (2001)
20. Prechelt, L.: Early stopping-but when? In: Neural Networks: Tricks of the trade, pp. 55–69. Springer (1998)
21. Radicati, S.: Email market, 2021-2025. The Radicati Group, Inc., Palo Alto, CA (2021)
22. Raffel, C., Ellis, D.P.: Feed-forward networks with attention can solve some long-term memory problems. arXiv preprint arXiv:1512.08756 (2015)
23. Rocktäschel, T., Grefenstette, E., Hermann, K.M., Kočiskỳ, T., Blunsom, P.: Reasoning about entailment with neural attention. arXiv preprint arXiv:1509.06664 (2015)
24. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE transactions on Signal Processing 45(11), 2673–2681 (1997)
25. Sharaff, A., Nagwani, N.K.: Identifying categorical terms based on latent dirichlet allocation for email categorization. In: Emerging Technologies in Data Mining and Information Security, pp. 431–437. Springer (2019)
26. Shroff, N., Sinhgala, A.: Email classification techniques—a review. Data Science and Intelligent Applications pp. 181–189 (2021)
27. Turc, I., Chang, M.W., Lee, K., Toutanova, K.: Well-read students learn better: On the importance of pre-training compact models. arXiv preprint arXiv:1908.08962 (2019)

28. Zhang, T.: Solving large scale linear prediction problems using stochastic gradient descent algorithms. In: Proceedings of the twenty-first international conference on Machine learning. p. 116 (2004)
29. Zhang, Y., Wallace, B.: A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. arXiv preprint arXiv:1510.03820 (2015)

Milena Šošić is a third-year PhD student under the supervision of dr Jelena Graovac at the University of Belgrade, Faculty of Mathematics. Her doctoral work explores the significance of different machine learning techniques in natural language processing and understanding fields with a focus on conversational text analysis. She holds a magister (mr) and graduation degrees in Computer Science, both from University of Belgrade, Faculty of Mathematics. She can be contacted at: pd202030@alas.matf.bg.ac.rs.

Jelena Graovac is Assistant Professor in the Department of Computer Science, Faculty of Mathematics, University of Belgrade. She received M.Sc. (2008, Computer Science) and Ph.D. (2014, Computer Science) degrees from the Faculty of Mathematics, University of Belgrade. The courses she taught at the University of Belgrade include Database Design, Information Systems, Introduction to Computer Organization and Architecture, Web Programming, Introduction to Programming, etc. (Faculty of Mathematics), and Intelligent Search (Intelligent Systems - the Ph.D. program of academic studies). Her research interests include Natural Language Processing, Information Retrieval, and Text Classification using Machine Learning and Knowledge-Based approaches. She co-authored many scientific papers as book chapters and articles in journals and conference proceedings.

Received: February 12, 2022; Accepted: Jun 01, 2022.