https://www.scirp.org/journal/jbise

J. Biomedical Science and Engineering, 2022, Vol. 15, (No. 1), pp: 6-25

The Effect of Speech Fragmentation and Audio Encodings on Automatic Parkinson's Disease Recognition

Dávid Sztahó[1], Attila Zoltán Jenei[1], István Valálik[2], Klára Vicsi[1]

[1]Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics, Budapest, Hungary; [2]Department of Neurosurgery, St. John's Hospital, Budapest, Hungary

Correspondence to: Dávid Sztahó, sztaho.david@vik.bme.hu; Attila Zoltán Jenei, jenei@tmit.bme.hu; István Valálik, valalik@parkinson.hu; Klára Vicsi, vicsi.klara@vik.bme.hu

Keywords: Parkinson's Disease, Speech, Support Vector Machine, Neural Network, i-Vector, x-Vector

Received: October 2, 2021 **Accepted:** January 7, 2022 **Published:** January 10, 2022

ABSTRACT

Parkinson's disease is a neurological disease which is incurable according to current clinical knowledge. Therefore, early detection and provision of appropriate treatment are of primary importance. Speech is one of the biomarkers that enable the detection of Parkinson's disease affection. Numerous researches are based on recordings from controlled environments; nonetheless fewer apply real circumstances. In the present study, three objectives were examined: recording fragmentation (paragraph, sentences, time-based), variable encodings (Pulse-Code Modulation [PCM], GSM-Full Rate [FR], G.723.1) and majority voting on 8 kHz records using multiple classifiers. Support Vector Machine (SVM), Long Short-Term Memory (LSTM), i-vector and x-vector classifiers were evaluated in contrast with SVM as baseline. The highest results in accuracy and F1-score were achieved using i-vector models. Although variable encodings generally caused decrease in Parkinson-disease recognition, decline was within 2% - 3% at best. Moreover, fragmentation did not yield a clear outcome though some classifiers performed with the very similar efficiency along the differently fragmented sets. Majority voting did produce a slight increase in classification performance compared to as if no aggregation is used.

1. INTRODUCTION

Parkinson's disease (PD) is the second most common neurological disorder which is affecting chiefly the elderly population. It is characterized by neuron death in the substantia-nigra area and accumulation of intracellular protein (α-synuclein) [1]. Its supreme syndrome consists of tremor, rigidity and bradykinesia often confounded with other forms of parkinsonism [2]. PD is an incurable disorder according to current clinical knowledge. Its symptoms and progress can only be restrained.

Several medicines can be applied to ease motor symptoms: dopamine precursors (e.g. Levodopa) or agonists (e.g. apomorphine) can be used to increase the amount of dopamine hormone. Furthermore, inhibiting the enzymes (Monoamine oxidase B (MAO-B) or Catechol-O-methyl transferase (COMT)) involved in dopamine metabolism can also be a way to maintain dopamine level [3]. In addition to medication, a number of therapies (such as movement and speech therapies) have been used to relieve symptoms and improve quality of life [4]. For example, the use of speech therapy can improve the patient's articulation and speech intelligibility. By moving similar muscle groups, the process of eating and swallowing can also be aided [5]. However, medications and therapies are necessary for the rest of the patient's life as they are only suitable for symptomatic induction. Therefore, it is key to recognize PD at an early stage and conduct medical care in the short term. Furthermore, tele-monitoring systems based on speech are getting more renowned due to the low cost [6].

Generally, patients with PD may experience monopitch, imprecise consonants or reduced loudness [7]. This allows space for speech analysis which is a noninvasive, rapid procedure. Examining speech in the early recognition or severity estimation of PD is already a wide area of research. Research articles have already included several speech tasks, such as sustained vowels [8], repetition of syllables (e.g., pa-ta-ka) [9, 10], read text [11], monologues [12]. Other modalities are also promising for the recognition of Parkinson's disease, such as movement, drawing, or electroencephalography (EEG). There are also multimodal solutions in the literature where multiple samples from a given person are examined [13]. In this research, the focus is on speech-based examination and recognition.

Most research reports methodologies that use the full length of speech recordings for analysis. This is the case when the patient reads a particular text. Afterwards, the task of the preprocessor algorithm is to extract features from the entire recordings in some way (e.g., moving window). With that methodology, one feature vector is resulted for one individual of the database. In this study, we examine the effect of segmenting the recordings into multiple durations (paragraph, sentences and time-based segments) on the performance of classification. With this augmentation, we can also produce more samples to train and test machine learning algorithms that may perform better with the cost of shorter parts to get useful information from. As far as we know, this is the first attempt to introduce such an investigation. Moreover, the aggregation of several decisions per speaker enables techniques, such as majority voting to improve the classification performance. This phenomenon is also investigated. By conducting such a research, it can create a basis whether a patient needs to read a longer text or just a part of it. That process may be time- and resource-efficient and also more convenient for the patient.

Secondly, many analyses rely on an artificial environment that is carefully controlled. Therefore, algorithms must deal with ideal, low-noise recordings where the only disturbance can be the presence of the disease. However, such rooms and quality microphones are least available in real life situations. This is the reason why there is a need to carry out research that takes real circumstances into account. There are already researches that illustrate processing of recordings coming from ordinary situations [14]. Their research covers certain background noises from different places, speech compression levels, codecs (e.g., GSM-full rate), and the impact of online media quality (e.g., Skype) among others. These effects were examined separately with the given feature set. Further results were obtained using telephone-recorded or simulated telephone-quality samples with binary classification accuracy ranging from 59% to 83% [15] [16]. The former research used 9783 subject (1483 PD) recordings collected with standard telephone systems under non-controlled conditions. Sustained /a/ vowels were used and 66.4% ± 1.8% accuracy was obtained. The latter paper compares samples recorded by professional microphone and actual telephones. They also simulated phone quality from the recordings of the professional microphone. Their results show primarily that the accuracy is near to the same (74% - 75%) either by using simulated quality or actual telephone recorded samples. They reached an accuracy up to 83% using a professional microphone. Therefore, the second objective of our research is the evaluation of models in a scenario where samples with encodings are used in mixed training and testing combinations: telephone communication system (using GSM-FR and G.723.1 codecs) and lossless PCM. As far as we know, this is the first attempt to examine

G.723.1 encoding, which is most commonly used in technical applications for voice communication over Voice over Internet Protocol (VoIP or IP telephony). GSM-FR is currently used in the majority of digital wireless telephone calls. Furthermore, it is unique to use all codecs in such a different combination for training and testing models. Such studies may provide a basis for examining a patient's telephone conversation even as an element of remote monitoring.

In this research, Hungarian-language recordings are used from healthy control and Parkinson's patients. Then, the recordings were split into smaller segments and also GSM-FR, G.723.1 codec versions were generated. Afterwards, SVM, LSTM, i-vector and x-vector classification models are proposed to recognize PD samples evaluated in the scenarios described earlier.

This paper is structured as the following: in the second Chapter, the applied methodology is presented. It includes the Hungarian Parkinson's Speech Dataset (HPSD), fragmentation of the recordings and the description of the classification tasks. In the third Chapter, the results are presented and in the fourth Chapter, conclusions are drawn. Finally, in the fifth Chapter, the key points of our research are summarized.

2. MATERIALS AND METHODS

The research reported in this article was based on the process shown in Figure 1. The recordings in the HPSD (detailed in Chapter 2.1) were fragmented while retaining the original samples. Beside the full length recordings (Paragraph) two additional sets were constructed: one that includes a few sentence long sample set (Sentences) and another includes a few second-long chunk set (Time-based).

Two additional codec versions (GSM-FR, G.723.1) of the three sets were generated while retaining the original PCM. In total, nine sets (three codecs for the three splitted sets) were available for further examination.

The nine speech sets were examined using four classification algorithms: SVM, LSTM, i-vector and x-vector. We conducted experiments for training and testing with samples of different durations and encodings. Furthermore, majority voting was performed for the sentences and time-based chunks using SVM algorithm.

2.1. HPSD Database

For the experiments, HPSD was used that contains recordings of 85 PD patients. The samples were recorded in two health institutes in Budapest: Semmelweis University and Virányos Clinic. The severity of PD patients was noted on the Hoehn and Yahr (H-Y) scale. It is a nonlinear scale ranging from 1 to 5,

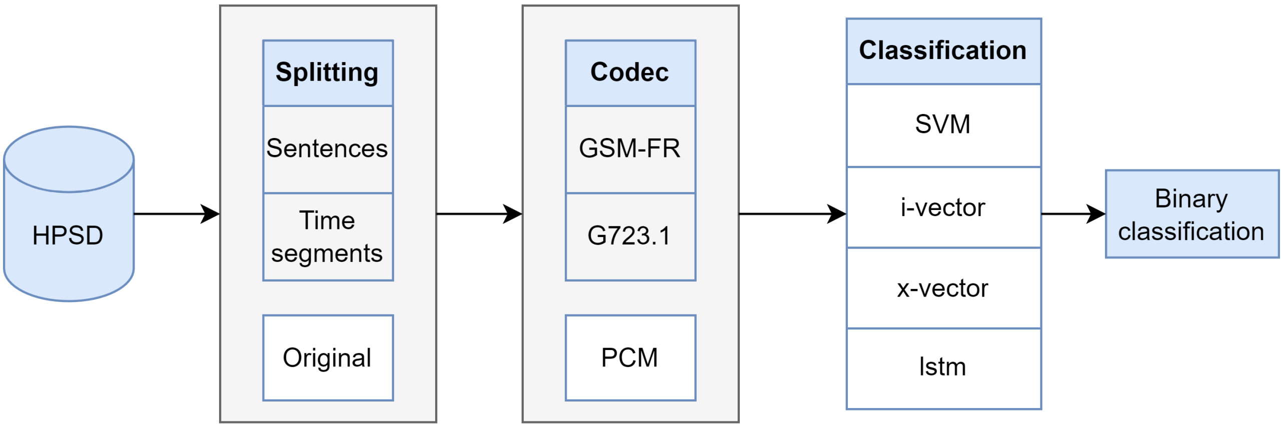

Figure 1. **Flowchart of the applied method: Speech database, splitting (paragraph, sentences, time-based), encoding-decoding (PCM, GSM-FR, G.723.1), classification (SVM, i-vector, x-vector, LSTM).**

where one indicates mild (unilateral) symptoms, while five stands for the most severe (bed or wheel-chair-bound) symptoms [17].

Overall, 85 healthy control (HC) and 85 PD recordings were collected using the tale of The North Wind and the Sun. From these, 86 males (mean age: 57.5 ± 15.0, mean H-Y for PD patients: 2.7 ± 1.1) and 84 females (mean age: 58.2 ± 11.3, mean H-Y for PD patients: 2.6 ± 1.1). HC subjects admitted that they had no known disease and were not during any medical treatment. All subjects signed a statement of consent when recordings were taken.

Recording was done with an external audio interface (Terratec 6fire USB) with PCM audio coding, 16 kHz sampling frequency and 16-bit quantization. A clip-on condenser microphone (Audio-Technica ATR3350) was used for recording in a quiet office environment.

2.2. Preprocessing

Recordings were normalized to peak amplitude. Samples were re-sampled at 8 kHz. Also, 16 kHz samples were retained to conduct a baseline model.

2.3. Variable Fragmentation

Recordings were split automatically by Praat [18] into two duration types (using available annotation): three-sentence long fragments (sentence boundaries were known) and 3 seconds-long chunks. The total number and mean length (standard deviation is included between brackets) of samples in each fragmentation set are shown in **Table 1** for the classes.

2.4. Variable Encodings

GSM encodings were done by Sound eXchange (SoX) for GSM-FR and a Matlab implementation for G.723.1 [19]. GSM-FR uses a bitrate of 13 kbit/s, G.723.1 uses a bitrate of 6.3 kbit/s (in Matlab implementation).

2.5. Classification Methods

2.5.1. Support Vector Machine

A linear and a radial basis function (rbf) kernel based SVM model were created in python (version 3.7.4) environment. In both cases, hyperparameters were set to default values (gamma: 1/feature number, C: 1.0, epsilon: 0.001).

Feature extraction for the SVM-based method was performed using the SurfBoard python package [20]. Instead using the original SurfBoard recipe, only 12 MFCCs (Mel-frequency cepstral coefficients) were extracted for later comparison of classification procedures. Mean and standard deviation statistics were accumulated into the final feature vector for each sound sample, along with their first derivative. A total of 48 features were extracted from each recording. Input features were normalized by scaling between −1 and 1.

Table 1. Number of samples and their length after splitting.

	Number of Recordings		Mean Length (Standard Deviation) of Recordings in Seconds	
	HC	PD	HC	PD
Paragraph	85	85	44.5 (±4.9)	59.0 (±23.2)
Sentences	340	340	10.4 (±2.6)	13.4 (±4.6)
Time-based	1029	1414	3.1 (±0.6)	3.2 (±0.3)

2.5.2. i-Vector and x-Vector

In speaker recognition and verification, i-vectors are state-of-the-art methods along with their deep learning variations [21]. In this model, factor analysis (FA) is used to compute a (originally) speaker- and session-dependent GMM supervector [22, 23] (created by concatenating the parameters of a background GMM with a large number of mixtures):

$$m_{s,h} = m_0 + Tw_{s,h}\,, \tag{1}$$

where m_0 is the GMM-UBM supervector, T is the speaker and channel factor, called total variability space and $w_{s,h}$~ N(0,1) are hidden variables, called total factors. The total factors are not observable, but can be estimated using FA. These total factors then can be used as features to a classifier, and came to be known as i-vectors (short for identity vector). The i-vector approach can be considered as a dimensionality reduction technique of the GMM supervector. Instead of speakers, other entities, such as diseases, can also be applied in the method, thus accomplishing classification tasks. The dimension of the i-vector in this study was set to 100.

Alternative to i-vectors, a deep learning based method (called x-vector) was also developed primarily for speaker verification [24]. It is based on a multiple layered DNN architecture (with fully connected layers) with different temporal context at each layer (which they call "frames"). Due to the wider temporal context, the architecture is called time-delay NN (TDNN). The TDNN embedding architecture can be seen in Figure 2 and Table 2.

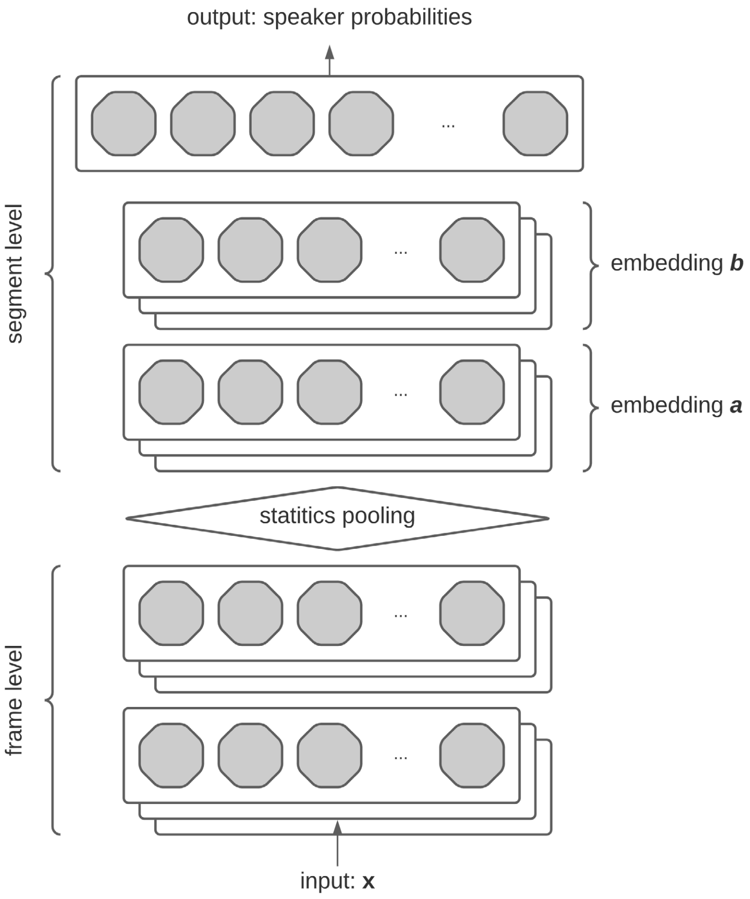

Figure 2. X-vector DNN embedding architecture in [24]. The two parts: frame level (with the 5 frame layers) and segment level (with segment 6, segment 7 and softmax).

Table 2. **X-vector DNN layer architecture [24]. It contains the layers, contexts and the input, output dimensions.**

Layer	Layer Context	Total Context	Input × Output
frame 1	[t − 2, t + 2]	5	120 × 512
frame 2	{t − 2, t, t + 2}	9	1536 × 512
frame 3	{t − 3, t, t + 3}	15	1536 × 512
frame 4	{t}	15	512 × 512
frame 5	{t}	15	512 × 1500
stats pooling	[0, T}	T	1500T × 3000
segment 6	{0}	T	3000 × 512
segment 7	{0}	T	512 × 512
softmax	{0}	T	512 × N

The first five layers operate on speech frames, with small temporal context centered at the current frame *t*. For example, the frame indexed as 3 sees a total of 15 frames, due to the temporal context of the earlier layers. After training with disease types as target vectors, the output of layer segment6 ("x-vector") is used as input to a classifier. The dimension of the x-vectors was set to 512.

The i-vector and x-vector implementations in this study followed the KALDI recipe of Snyder in which 12 MFCCs were used as input features.

Probabilistic linear discriminant analysis (PLDA) [25] was used for scoring i-vector and x-vector representation of samples. Classification was achieved by selecting the class (HC or PD) that had a higher score resulting from the algorithm.

2.5.3. Long Short-Term Memory

A deep learning classification method following [26] was also evaluated in the current study. The DL architecture consists of two parts: an autoencoder and LSTM part. It learns a feature representation and a disease specific part performs classification in a muti-learning setup (**Figure 3**).

12 MFCCs of the audio samples were used as input features with 25 ms calculation window and 10 ms time step. The method was implemented in Tensorflow 2.1. Complexity of the network was set to the values described in the original study, summarized in **Table 3**.

2.5.4. Evaluation Setup

5-fold cross validation was applied during the experiments to split the samples into training and test sets. These sets never contained samples from the same speaker (cross-validation splitting was done according to speaker ids). No model optimization was performed, default values described previously were applied as hyperparameters; hence, no development set was available due to the relatively low number of samples. All hyperparameters were chosen according to the referenced studies. Classifier methods were evaluated by accuracy and F1-score, referred as "Acc." and "F1" in the later Tables describing results. For each test case, more detailed results are shown in Appendix depicting also specificity and sensitivity.

Three experimental scenarios were evaluated. A baseline performance level was created by linear and rbf kernel SVM using samples without resampling (16 kHz) and applying all three fragmentation sets (paragraph, sentences, time-based).

The possible performance variation of SVM, LSTM, i-vector and x-vector methods due to different encodings and fragmentation sets were evaluated using varying codecs and fragmentation sets applied as

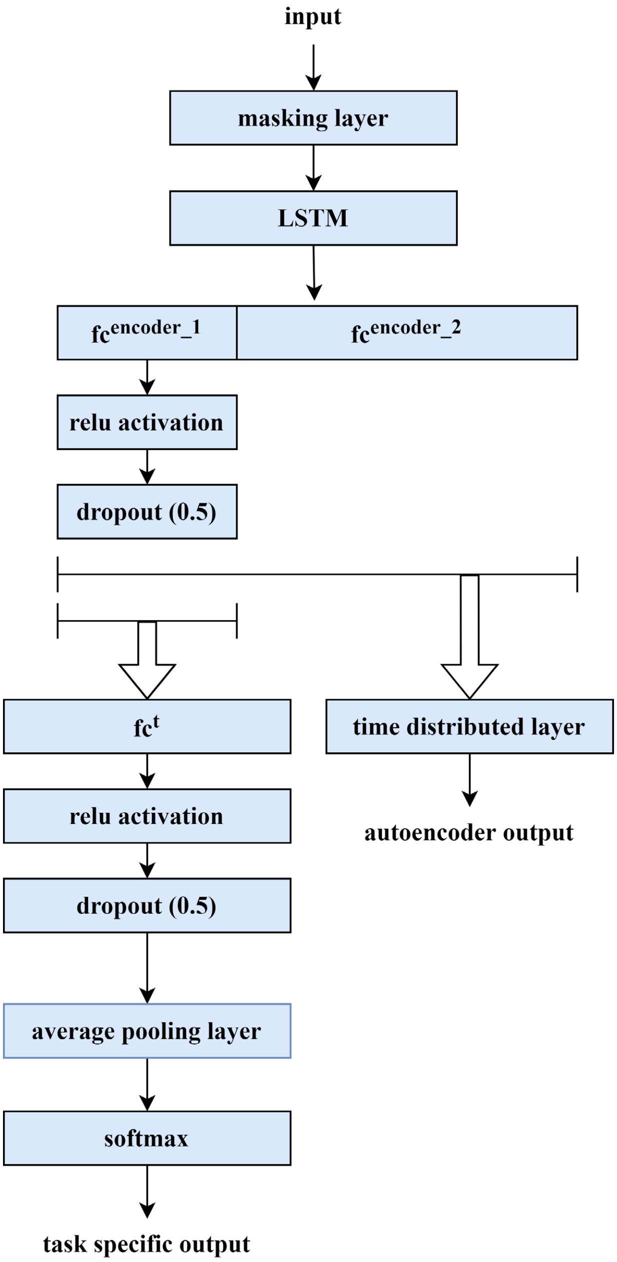

Figure 3. LSTM-auto encoder-multitask learning network in [26]. The upper part serves the task of the autoencoder while the lower part performs the classification.

Table 3. Number of units in DL layers.

Layer Name	Units
LSTM	100
$fc^{encoder\_1}$	30
$fc^{encoder\_2}$	30
fc^{t}	200
time distributed layer	128

train and test sets.

A majority voting scheme was also examined with linear and rbf kernel SVM in which the decision for each speaker was determined by selecting the most frequently occurring class label of the given speaker's samples.

3. RESULTS

3.1. Baseline Results of 16 kHz PCM Encodings

Linear and rbf SVM trained with the original 16 kHz PCM version of the samples (without resampling) was used as an overall baseline. Results on the three fragmentation sets are shown in Table 4. Acc. and F1 abbreviation refer to accuracy and F1-score, respectively.

Using full length recordings (paragraph) 85.3% accuracy and 85.4% F1-score were achieved with linear kernel SVM. By splitting paragraphs into sentences, nearly the same results can be observed in both accuracy (from 85.3% to 85.0%) and F1-score (from 85.4% to 85.2%). However, time-based chunking resulted in a slight decrease (from 85.0% to 83.5%) in accuracy and a slight improvement in F1-score (from 85.2% to 86.1%) compared to sentences results.

With the algorithm of rbf kernel SVM, the results are nearly the same as with the linear kernel. The paragraph case performed a bit lower while the outcomes increased slightly with sentences. The time-based training and testing resulted about the same accuracy and F1-score as the linear kernel SVM did.

An extended version of result table of the baseline experiment can be seen in Table A1, **Appendix A**.

3.2. Variable Fragmentation Results of 8 kHz PCM Encodings

In order to evaluate the effects of samples with various durations on the classification performance, trials were run by applying fragmentation sets as various train and test sets combinations. The results can be seen in Table 5 for all classifier models.

In case of paragraph training, testing with recordings of same duration yielded the best accuracy and F1-score values for most of the classifiers. Of these, rbf kernel SVM had achieved the highest accuracy (89.4%) and F1-score (89.7%). It can also be seen that if shorter recordings are used for testing, the results generally deteriorated. It is noteworthy that this change was within 1% - 2% for i-vector while at most 15% for LSTM.

For training with samples fragmented into sentences it can be seen that the best results were obtained when the test was made with paragraphs (i-vector, x-vector, SVM-linear). However, the difference is not significant. Within this, the i-vector resulted in the highest accuracy (90.0%) and F1-score (90.3%). It also ensues from these outcomes that training and testing with sentences did not achieve the result that training with sentences and testing with paragraphs (only with SVMs).

By training with time-based segments and testing with paragraphs, i-vector yielded the best results

Table 4. Baseline results (accuracy and F1-score) of 16 kHz PCM recordings with linear and rbf kernel SVM.

Trained/Tested on	rbf kernel		linear kernel	
	Acc.	F1	Acc.	F1
Paragraph/Paragraph	84.7%	84.5%	85.3%	85.4%
Sentences/Sentences	86.0%	86.0%	85.0%	85.2%
Time-based/Time-based	83.8%	86.2%	83.5%	86.1%

Table 5. Results of 8 kHz PCM recordings of different durations using multiple classification model.

Trained/ Tested on	i-vector		x-vector		LSTM		SVM-linear		SVM-rbf	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
P/P	88.8%	89.4%	84.1%	83.6%	81.2%	80.2%	84.1%	84.6%	89.4%	89.7%
P/S	88.5%	89.1%	79.1%	78.0%	78.2%	76.9%	74.7%	78.9%	80.7%	83.1%
P/T	88.2%	90.2%	74.2%	75.9%	73.7%	66.9%	74.9%	81.6%	79.5%	84.2%
S/P	90.0%	90.3%	80.6%	80.9%	81.2%	81.4%	84.7%	84.3%	80.0%	77.3%
S/S	88.8%	89.4%	79.3%	78.9%	80.0%	79.8%	84.6%	84.6%	86.2%	86.3%
S/T	88.4%	90.3%	75.2%	77.6%	74.0%	70.1%	79.1%	81.1%	81.3%	83.2%
T/P	89.4%	89.8%	76.5%	76.5%	83.5%	82.7%	85.9%	85.9%	82.4%	82.6%
T/S	89.0%	89.2%	77.5%	77.1%	84.7%	84.2%	82.4%	83.7%	85.0%	85.9%
T/T	87.4%	89.4%	74.7%	77.0%	81.0%	77.6%	82.6%	85.5%	84.6%	87.0%

In the first column, the abbreviations are: P—Paragraph, S—Sentences, T—Time-based.

(accuracy: 89.4%, F1-score: 89.8%). Testing with sentences and time-based segments, the i-vector also achieved the highest metrics values. Moreover, all classifiers reached higher performance than 80% both in accuracy and F1-score except the x-vector and LSTM. These are deep learning based algorithms that may need more data to ensure robust classification.

The slight increase of performance when trained on smaller audio chunks (sentences and time-based) may result from the increased number of training data for which all machine learning applications are sensitive. However, the small amount of change in accuracy and F1-score imply that in-formation regarding Parkinson speech disorder is also present in smaller speech segments.

The extended version of Table 5 can be found in **Appendix A** named as Table A2.

3.3. Variable Encoding Results of 8 kHz Recordings

In real-world scenarios, recordings can be affected by channel mismatches, such as differences in encodings of mobile phone audio signal. To investigate the performance degradation in such cases, the three different encodings were applied in all train and test sets combinations. The obtained results are shown in Tables 6-8 for the fragmentation sets separately. The rows show the results of different training and testing codec combinations, the columns show the results of the different classifiers.

Using paragraphs (Table 6), the i-vector achieved the highest accuracy (90.6%) and F1-score (90.4%) when trained with PCM and tested with G.723.1. Similar high results to i-vector were obtained using SVM with radial basis function as kernel in case of PCM/GSM-FR (accuracy: 87.6%, F1-score: 87.1%). In the case of the other classifier, a larger fluctuation can be realized. The x-vector and LSTM show a decrease in performance when the models trained with PCM and tested with G.723.1 instead of GSM-FR. However, the linear kernel SVM provides nearly the same performance (even higher in the case of GSM-FR/PCM than in the case of PCM/GSM-FR). A similar degradation was perceived generally when training was done with codecs, and tested was done with PCM.

Using sentence splitted recordings (Table 7), the i-vector achieved the top results (accuracy: 90.0% - 86.5%, F1-score: 90.2% - 87.5%). Moreover, training with G.723.1 and testing with PCM codec yielded nearly the same accuracy and F1-score as PCM/GSM-FR using i-vector and linear kernel SVM. A decrease is also observed when training was done with GSM codecs and the testing was done with PCM in many cases. However, the SVMs', LSTM's and i-vector's decreases are narrow along the cases.

Table 6. Results of variable encodings on PARAGRAPH 8 kHz recordings using multiple classification models.

Trained/ Tested on	i-vector		x-vector		LSTM		SVM-linear		SVM-rbf	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
PCM/GSM-FR	87.1%	88.3%	81.2%	79.5%	81.2%	80.0%	80.6%	81.6%	87.6%	87.1%
PCM/G.723.1	90.6%	90.4%	78.2%	76.7%	77.6%	72.9%	80.6%	79.2%	84.1%	82.8%
GSM-FR/PCM	87.6%	87.7%	78.8%	79.3%	82.4%	83.0%	85.9%	86.7%	86.5%	87.3%
G.723.1/PCM	87.1%	87.8%	74.7%	73.6%	80.6%	82.2%	75.9%	80.0%	81.8%	83.8%

Table 7. Results of variable encodings on SENTENCES 8 kHz recordings using multiple classification models.

Trained/ Tested on	i-vector		x-vector		LSTM		SVM-linear		SVM-rbf	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
PCM/GSM-FR	86.5%	87.5%	80.6%	80.2%	82.1%	82.0%	83.1%	83.4%	85.1%	85.6%
PCM/G.723.1	90.0%	90.2%	76.5%	76.5%	79.3%	77.0%	80.4%	82.2%	84.3%	85.4%
GSM-FR/PCM	89.1%	89.5%	75.9%	76.6%	80.9%	81.6%	83.7%	83.7%	83.5%	83.0%
G.723.1/PCM	87.6%	87.9%	75.3%	74.4%	79.3%	80.2%	80.3%	78.0%	83.5%	82.4%

Table 8. Results of variable encodings on TIME-BASED 8 kHz recordings using multiple classification models.

Trained/ Tested on	i-vector		x-vector		LSTM		SVM-linear		SVM-rbf	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
PCM/GSM-FR	87.2%	89.3%	80.0%	79.3%	81.0%	77.7%	83.1%	86.4%	84.2%	87.1%
PCM/G.723.1	86.1%	87.9%	76.5%	76.5%	81.0%	76.0%	82.6%	85.9%	82.7%	85.9%
GSM-FR/PCM	87.8%	89.4%	75.9%	76.0%	82.6%	78.8%	81.2%	83.5%	82.4%	84.4%
G.723.1/PCM	85.9%	87.7%	75.3%	72.7%	80.1%	77.4%	78.9%	80.0%	81.0%	82.6%

For time-based segments (Table 8), the best results came with the i-vector as well (accuracy: 87.8% - 85.9%, F1-score: 89.4% - 87.7%). A small decrease can be observed along all the classifiers examining different cases.

Looking at Tables 6-8, and comparing it to Table 4, performance changes can be observed and the trend of metrics' increase/decrease is not that clear so far. This implies that using these coded speech samples does not have an unequivocal effect on trained models. However, choosing such models, this performance changes can be minimal. The i-vector approach, due to its dimension reduction method, can be considered as a robust machine learning algorithm in present case. The second machine learning solution would be the rbf kernel SVM with narrow decrease and high performance.

The expended result tables of Tables 6-8 can be found in **Appendix A** as Tables A3-A5, respectively.

3.4. Majority Voting on 8 kHz SVM Results

Due to the fragmentation of recordings, multiple samples were available for each speaker. A way of fusing decisions made for a speaker is majority voting. The result of majority voting for time-based and sentence fragmentation experiments can be seen in Table 9 for SVM with linear and rbf (MV: majority voting, no MV: without majority voting).

It can be deduced from Table 9 that improvement was achieved with majority voting on both time-based and sentence sets compared to not aggregating decisions for the subjects. The accuracy of SVM with rbf kernel changed from 84.6% to 87.1% and the F1-score changed from 87.0% to 87.8% in case of time-based segments. Using sentences, the accuracy from 86.2% rose to 87.6% while the F1-score increased from 86.3% to 88.1%. Using linear kernel, accuracy (time-based: 82.6% to 87.1%, sentences: 84.6% to 87.7%) and F1-score (time-based: 85.5% to 87.8%, sentences: 84.6% to 87.9%) also improved for both time-based and sentence chunking experiments.

Using paragraphs to train and test SVM with rbf, 89.4% accuracy and 89.7% F1-score were achieved (Table 5). It can be seen that higher performance cannot be achieved by majority voting in this case. However, improvement is observed with majority voting rather than paragraph method (accuracy: 84.1%, F1-score: 84.6%) using linear kernel SVM.

The extended version of majority voting result table can be found in Table A6, **Appendix A**.

4. DISCUSSION

The results obtained with the baseline models (accuracy: 85.3% - 83.5%) can be compared with other findings in the literature. Based on a survey [27], it reports accuracy above 66% using SVM. Moreover, most research achieved over 85% accuracy. The comparison should also be treated with caution, as the speech databases applied were different.

Table 4 and Table 5 show a few percent performance difference between the 8 and 16 kHz recordings with the SVM classifier. Specifically, 8 kHz recordings resulted in a slight decrease (max. −2%) for the

Table 9. **Results of majority voting using SVM with linear and rbf kernel on 8 kHz samples.**

Experiment			rbf kernel		linear kernel	
	Training On	Testing On	Acc.	F1	Acc.	F1
no MV	Time-based	Time-based	84.6%	87.0%	82.6%	85.5%
MV	Time-based	Time-based	87.1%	87.8%	87.1%	87.8%
no MV	Sentences	Sentences	86.2%	86.3%	84.6%	84.6%
MV	Sentences	Sentences	87.6%	88.1%	87.7%	87.9%

linear kernel, while for the rbf kernel, they resulted in a slight increase (max +5%). Using multiple classification models, better results were achieved in many cases than baseline outcomes. This is because 16 kHz recordings may contain interfering noises, which may impair recognition. Moreover, the different classifiers process information differently that can be resulted in distinct performances as well.

The experiments of Chapter 3.2 do not give a clear result as to whether it is worthwhile to use recordings of different durations for training and testing. On the other hand, it seems that i-vector, x-vector and SVM kept their result range narrow along with the different training/testing layouts.

In the case of deep learning solutions, it may be worth noting that the full length recordings produced results comparable the other classifiers. On the other hand, the performance of these classifiers typically deteriorated when splitted segments were used. In this case, despite the increase in the number of training data, the recordings no longer contained as much information as it would be efficient for high performance training. A possible way to increase the performance of x-vector is to use a pre-trained x-vector extractor trained on a large dataset of a different domain (such as speaker verification). However, there are doubts about this method precisely because of domain differences. Testing this is beyond the scope and aim of this paper, but it would be worthwhile to investigate the matter. The i-vector approach however had small variation in performance taking all fragmentations into consideration. It was already experienced that in case of a small number of samples, i-vector outperforms x-vector [28].

In terms of encoding experiments, the classifiers performed worse when telephone coded recordings were used for testing. The results were even more deteriorated when the model was trained with GSM encoded samples. However, it should be noted that there is a little change between telephone coded and PCM recordings based on manual inspection of spectrograms. The variation in performance is also not significant along multiple classifiers. Based on these findings, it is not necessarily true that prediction on low quality recordings (8 kHz sampling rate, GSM encoding with low bitrate) cannot be performed using a model trained with recordings acquired in a high quality setup (16 kHz sampling rate, lossless coding). There is a moderate performance loss however, which must be taken into consideration.

If multiple recordings per speaker are available, majority voting may give opportunity to improve the classification performance rather than using multiple decisions per subject separately. This is because there may be some samples with a difficulty to distinguish between PD and HC. Moreover, this method did exceed the results of training and testing with complete paragraphs using SVM with linear kernel.

5. CONCLUSION

In the present study, recognition of Parkinson's disease was performed. Multiple phenomena were examined: the effect of speech recording fragmentation and mixed use of variable encodings. Full length recordings (paragraphs, short read tale) were divided into sentences and short time-based chunks. Multiple classification models were applied on these sets separately and crosswise. Effect of telephone codecs beside lossless PCM (using samples resampled to 8 kHz) was also considered throughout the classification performance. The results were compared to a baseline outcome of SVM with linear and rbf kernel function. Finally, majority voting was carried out where multiple samples had been available per subject.

Similar results can be obtained on the 8 kHz recordings than on the 16 kHz ones using linear kernel SVM. Changing only the kernel of SVM, the result improved. Using x-vector and i-vector, the result was increased even more. In addition, the separate and crosswise experiments did not give a clear result whether it is worthwhile to do classification using either smaller segments or crosswise training/testing layout. However, the outputs of certain models are slightly altered in different training/testing designs. Using telephone codecs, the change in recording quality is reflected in the outcomes of the algorithms. As well, certain machine learning models could handle telephone codecs with narrow deviations in performance.

Finally, the use of majority voting was able to marginally increase the performance of the classifier. Moreover, better results were achieved as using paragraphs in case of linear kernel SVM. It can be concluded that majority voting may improve the performance in case of multiple samples per subject. Furthermore, GSM encodings does not deteriorate the performance of the models to be unusable. Especially, certain classifiers can retain the same result with GSM coded samples than with purely PCM ones.

Based on the results presented here, two main statements can be made. 1) The patient's speech over a telephone call could be also examined with such an algorithm (that trained on GSM coded samples). Thus, the procedure would not necessarily require a personal presence and call quality would also be taken into account. This may even play a role in monitoring patients in a treatment follow-up or pre-screening subjects using cheap devices and protocol. 2) It is still an open question, what is the speech length sufficient for an evaluation. It may be enough to record only a part of the read text instead of the full version. This is more time efficient and convenient for both the doctor and the patient.

ACKNOWLEDGEMENTS

Project no. K128568 has been implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the K_18 funding scheme. The research was partly funded by the CELSA (CELSA/18/027) project titled: "Models of Pathological Speech for Diagnosis and Speech Recognition".

CONFLICTS OF INTEREST

The authors declare no conflicts of interest regarding the publication of this paper.

REFERENCES

1. Poewe, W., Seppi, K., Tanner, C.M., Halliday, G.M., Brundin, P., Volkmann, J., Schrag, A.-E. and Lang, A.E. (2017) Parkinson Disease. *Nature Reviews. Disease Primers*, **3**, Article No. 17013. https://doi.org/10.1038/nrdp.2017.13

2. Balestrino, R. and Schapira, A.H.V. (2020) Parkinson Disease. *European Journal of Neurology*, **27**, 27-42. https://doi.org/10.1111/ene.14108

3. Michael, E.J. and Matthew, J.F. (2017) Current Approaches to the Treatment of Parkinson's Disease. *Bioorganic & Medicinal Chemistry Letters*, **27**, 4247-4255. https://doi.org/10.1016/j.bmcl.2017.07.075

4. Gage, H. and Storey, L. (2004) Rehabilitation for Parkinson's Disease: A Systematic Review of Available Evidence. *Clinical Rehabilitation*, **18**, 463-482. https://doi.org/10.1191/0269215504cr764oa

5. Sapir, S., Ramig, L. and Fox, C. (2008) Speech and Swallowing Disorders in Parkinson Disease. *Current Opinion in Otolaryngology & Head and Neck Surgery*, **16**, 205-210. https://doi.org/10.1097/MOO.0b013e3282febd3a

6. Klumpp, P., Janu, T., Arias-Vergara, T., Vásquez-Correa, J.C., Orozco-Arroyave, J.R. and Nöth, E. (2017) Apkinson—A Mobile Monitoring Solution for Parkinson's Disease. *Interspeech* 2017, Stockholm, 20-24 August 2017, 1839-1843. https://doi.org/10.21437/Interspeech.2017-416

7. Vasquez-Correa, J.C., Arias-Vergara, T., Orozco-Arroyave, J.R., Eskofier, B., Klucken, J. and Noth, E. (2019) Multimodal Assessment of Parkinson's Disease: A Deep Learning Approach. *IEEE Journal of Biomedocal and Health Informatics*, **23**, 1618-1630. https://doi.org/10.1109/JBHI.2018.2866873

8. Dromey, C., Ramig, L.O. and Johnson, A.B. (1995) Phonatory and Articulatory Changes Associated with Increased Vocal Intensity in Parkinson Disease: A Case Study. *Journal of Speech, Language, and Hearing Research*, **38**, 751-764. https://doi.org/10.1044/jshr.3804.751

9. Novotný, M., Rusz, J., Čmejla, R. and Růžička, E. (2014) Automatic Evaluation of Articulatory Disorders in Parkinson's Disease. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **22**, 1366-1378. https://doi.org/10.1109/TASLP.2014.2329734

10. Sztahó, D., Valálik, I. and Vicsi, K. (2019) Parkinson's Disease Severity Estimation on Hungarian Speech Using Various Speech Tasks. 2019 *International Conference on Speech Technology and Human-Computer Dialogue*

(*SpeD*), Timisoara, 10-12 October 2019, 1-6. https://doi.org/10.1109/SPED.2019.8906277

11. Kiss, G., Takács, A.B., Sztahó, D. and Vicsi, K. (2018) Detection Possibilities of Depression and Parkinson's Disease Based on the Ratio of Transient Parts of the Speech. 9*th IEEE International Conference on Cognitive Infocommunications* (*CogInfoCom*), Budapest, 22-24 August 2018, 165-168. https://doi.org/10.1109/CogInfoCom.2018.8639901

12. Sztahó, D., Tulics, M.G., Vicsi, K. and Valálik, I. (2017) Automatic Estimation of Severity of Parkinson's Disease Based on Speech Rhythm Related Features. 8*th IEEE International Conference on Cognitive Infocommunications* (*CogInfoCom*), Debrecen, 11-14 September 2017, 11-16. https://doi.org/10.1109/CogInfoCom.2017.8268208

13. Jinee, G., Padmavati, K. and Trilok, C.A. (2020) Classification, Prediction, and Monitoring of Parkinson's Disease Using Computer Assisted Technologies: A Comparative Analysis. *Engineering Applications of Artificial Intelligence*, **96**, Article ID: 103955. https://doi.org/10.1016/j.engappai.2020.103955

14. Vasquez-Correa, J.C., Serra, J., Orozco-Arroyave, J.R., Vargas-Bonilla, J.F. and Noth, E. (2017) Effect of Acoustic Conditions on Algorithms to Detect Parkinson's Disease from Speech. *IEEE International Conference on Acoustics*, *Speech and Signal Processing* (*ICASSP*), New Orleans, 5-9 March 2017, 5065-5069. https://doi.org/10.1109/ICASSP.2017.7953121

15. Arora, S., Baghai-Ravary, L. and Tsanas, A. (2019) Developing a Large Scale Population Screening Tool for the Assessment of Parkinson's Disease Using Telephone-Quality Voice. *The Journal of the Acoustical Society of America*, **145**, 2871-2884. https://doi.org/10.1121/1.5100272

16. Jeancolas, L., Mangone, G., Corvol, J.-C., Vidailhet, M., Lehéricy, S., Benkelfat, B.-E., Benali, H. and Petrovska-Delacrétaz, D. (2019) Comparison of Telephone Recordings and Professional Microphone Recordings for Early Detection of Parkinson's Disease, Using Mel-Frequency Cepstral Coefficients with Gaussian Mixture Models. *Interspeech* 2019, Graz, 15-19 September 2019, 3033-3037. https://doi.org/10.21437/Interspeech.2019-2825

17. Hoehn, M.M. and Yahr, M.D. (1967) Parkinsonism: Onset, Progression and Mortality. *Neurology*, **17**, 427-442. https://doi.org/10.1212/WNL.17.5.427

18. Boersma, P. and van Heuven, V. (2001) Praat, a System for Doing Phonetics by Computer. *GLOT International*, **5**, 341-345. https://www.fon.hum.uva.nl/paul/papers/speakUnspeakPraat_glot2001.pdf

19. Peter, K. G.723.1 Speech Coder and Decoder. https://se.mathworks.com/matlabcentral/fileexchange/24755-g-723-1-speech-coder-and-decoder

20. Lenain, R., Weston, J., Shivkumar, A. and Fristed, E. (2020) Surfboard: Audio Feature Extraction for Modern Machine Learning. *Interspeech*, Shanghai, 25-29 October 2020, 2917-2921. https://doi.org/10.21437/Interspeech.2020-2879

21. Dehak, N., Kenny, P.J., Dehak, R., Dumouchel, P. and Ouellet, P. (2011) Front-End Factor Analysis for Speaker Verification. *IEEE Transactions on Audio*, *Speech*, *and Language Processing*, **19**, 788-798. https://doi.org/10.1109/TASL.2010.2064307

22. Campbell, W.M., Sturim, D.E. and Reynolds, D.A. (2006) Support Vector Machines Using GMM Supervectors for Speaker Verification. *IEEE Signal Processing Letters*, **13**, 308-311. https://doi.org/10.1109/LSP.2006.870086

23. Reynolds, D.A. and Rose, R.C. (1995) Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models. *IEEE Transactions on Speech and Audio Processing*, **3**, 72-83. https://doi.org/10.1109/89.365379

24. Snyder, D., Garcia-Romero, D., Sell, G., Povey, D. and Khudanpur, S. (2018) X-Vectors: Robust DNN Embeddings for Speaker Recognition. *IEEE International Conference on Acoustics*, *Speech and Signal Processing*

(*ICASSP*), Calgary, 15-20 April 2018, 5329-5333. https://doi.org/10.1109/ICASSP.2018.8461375

25. Ioffe, S. (2006) Probabilistic Linear Discriminant Analysis. In: Leonardis, A., Bischof, H. and Pinz, A., Eds., *Computer Vision—ECCV* 2006, Lecture Notes in Computer Science, Vol. 3954, Springer, Berlin, 531-542. https://doi.org/10.1007/11744085_41
26. Sztahó, D., Kiss, G. and Tulics, M.G. (2021) Deep Learning Solution for Pathological Voice Detection Using LSTM-Based Autoencoder Hybrid with Multi-Task Learning. *Proceedings of the* 14*th International Joint Conference on Biomedical Engineering Systems and Technologies* (*BIOSTEC*), Vienna, 11-13 February 2021, 135-141. https://doi.org/10.5220/0010193101350141
27. Bind, S., Tiwari, A.K. and Sahani, A.K. (2015) A Survey of Machine Learning Based Approaches for Parkinson Disease Prediction. *International Journal of Computer Science and Information Technologies*, **6**, 1648-1655. https://www.ijcsit.com/docs/Volume%206/vol6issue02/ijcsit20150602163.pdf
28. Sarkar, A.K., Matrouf, D., Bousquet, P.M. and Bonastre, J.-F. (2001) Study of the Effect of i-Vector Modeling on Short and Mismatch Utterance Duration for Speaker Verification. 13*th Annual Conference of the International Speech Communication Association*, Portland, 9-13 September 2012, 2662-2665.

APPENDIX A

Table A1. Extended results table of the baseline experiments using SVM algorithms.

	Trained/Tested on	Paragraph/Paragraph	Sentences/Sentences	Time-based/Time-based
rbf kernel	Spec.	85.9%	85.9%	78.3%
	Sen.	83.5%	86.2%	87.8%
	Pred.	85.5%	85.9%	84.8%
	Acc.	84.7%	86.0%	83.8%
	F1	84.5%	86.0%	86.2%
linear kernel	Spec.	84.7%	83.5%	77.8%
	Sen.	85.9%	86.5%	87.7%
	Pred.	84.9%	84.0%	84.5%
	Acc.	85.3%	85.0%	83.5%
	F1	85.4%	85.2%	86.1%

In the second column, the abbreviations are: Spec.—Specificity, Sen.—Sensitivity, Prec.—Precision, Acc.—Accuracy, F1—F1-score.

Table A2. Extended results table of the recordings' length variation experiments.

	Trained/Tested on	P/P	P/S	P/T	S/P	S/S	S/T	T/P	T/S	T/T
i-vector	Spec.	83.5%	83.2%	80.7%	87.1%	83.5%	81.1%	85.9%	86.8%	81.7%
	Sen.	94.1%	93.8%	93.7%	92.9%	94.1%	93.6%	92.9%	91.2%	91.5%
	Prec.	85.1%	84.8%	86.9%	87.8%	85.1%	87.2%	86.8%	87.3%	87.3%
	Acc.	88.8%	88.5%	88.2%	90.0%	88.8%	88.4%	89.4%	89.0%	87.4%
	F1	89.4%	89.1%	90.2%	90.3%	89.4%	90.3%	89.8%	89.2%	89.4%
x-vector	Spec.	87.1%	84.4%	79.6%	78.8%	80.9%	77.0%	76.5%	79.1%	77.0%
	Sen.	81.2%	73.8%	70.3%	82.4%	77.6%	74.0%	76.5%	75.9%	73.1%
	Prec.	86.3%	82.6%	82.6%	79.5%	80.2%	81.5%	76.5%	78.4%	81.4%
	Acc.	84.1%	79.1%	74.2%	80.6%	79.3%	75.2%	76.5%	77.5%	74.7%
	F1	83.6%	78.0%	75.9%	80.9%	78.9%	77.6%	76.5%	77.1%	77.0%
LSTM	Spec.	78.5%	75.3%	75.2%	81.9%	79.3%	78.9%	80.6%	82.8%	83.9%
	Sen.	84.4%	82.0%	71.1%	80.5%	80.7%	67.9%	87.0%	86.9%	77.0%
	Prec.	76.5%	72.4%	63.2%	82.4%	78.8%	72.4%	78.8%	81.8%	78.1%
	Acc.	81.2%	78.2%	73.7%	81.2%	80.0%	74.0%	83.5%	84.7%	81.0%
	F1	80.2%	76.9%	66.9%	81.4%	79.8%	70.1%	82.7%	84.2%	77.6%

Continued

SVM-linear	Spec.	81.2%	54.7%	45.0%	87.1%	84.1%	81.8%	85.9%	74.1%	74.7%
	Sen.	87.1%	94.7%	96.6%	82.4%	85.0%	77.2%	85.9%	90.6%	88.4%
	Prec.	82.2%	67.6%	70.7%	86.4%	84.3%	85.4%	85.9%	77.8%	82.8%
	Acc.	84.1%	74.7%	74.9%	84.7%	84.6%	79.1%	85.9%	82.4%	82.6%
	F1	84.6%	78.9%	81.6%	84.3%	84.6%	81.1%	85.9%	83.7%	85.5%
SVM-rbf	Spec.	87.1%	66.8%	58.2%	91.8%	85.3%	83.7%	81.2%	78.5%	78.4%
	Sen.	91.8%	94.7%	94.9%	68.2%	87.1%	79.6%	83.5%	91.5%	89.1%
	Prec.	87.6%	74.0%	75.7%	89.2%	85.5%	87.0%	81.6%	81.0%	85.0%
	Acc.	89.4%	80.7%	79.5%	80.0%	86.2%	81.3%	82.4%	85.0%	84.6%
	F1	89.7%	83.1%	84.2%	77.3%	86.3%	83.2%	82.6%	85.9%	87.0%

In the second column, the abbreviations are: Spec.—Specificity, Sen.—Sensitivity, Prec.—Precision, Acc.—Accuracy, F1—F1-score. In the first row, the abbreviations are: P—Paragraph, S—Sentences, T—Time-based.

Table A3. Extended results table of the variable encodings on PARAGRAPH experiments.

Trained/Tested on		PCM/GSM-FR	PCM/G.723.1	GSM-FR/PCM	G.723.1/PCM
i-vector	Spec.	76.5%	92.9%	87.1%	81.2%
	Sen.	97.6%	88.2%	88.2%	92.9%
	Prec.	80.6%	92.6%	87.2%	83.2%
	Acc.	87.1%	90.6%	87.6%	87.1%
	F1	88.3%	90.4%	87.7%	87.8%
x-vector	Spec.	89.4%	84.7%	76.5%	78.8%
	Sen.	72.9%	71.8%	81.2%	70.6%
	Prec.	87.3%	82.4%	77.5%	76.9%
	Acc.	81.2%	78.2%	78.8%	74.7%
	F1	79.5%	76.7%	79.3%	73.6%
LSTM	Spec.	77.9%	70.4%	84.8%	87.1%
	Sen.	85.3%	92.7%	80.2%	76.0%
	Prec.	75.3%	60.0%	85.9%	89.4%
	Acc.	81.2%	77.6%	82.4%	80.6%
	F1	80.0%	72.9%	83.0%	82.2%

Continued

SVM-linear	Spec.	75.3%	87.1%	80.0%	55.3%
	Sen.	85.9%	74.1%	91.8%	96.5%
	Prec.	77.7%	85.1%	82.1%	68.3%
	Acc.	80.6%	80.6%	85.9%	75.9%
	F1	81.6%	79.2%	86.7%	80.0%
SVM-rbf	Spec.	91.8%	91.8%	80.0%	69.4%
	Sen.	83.5%	76.5%	92.9%	94.1%
	Prec.	91.0%	90.3%	82.3%	75.5%
	Acc.	87.6%	84.1%	86.5%	81.8%
	F1	87.1%	82.8%	87.3%	83.8%

In the second column, the abbreviations are: Spec.—Specificity, Sen.—Sensitivity, Prec.—Precision, Acc.—Accuracy, F1—F1-score. In the first row, the abbreviations are: P—Paragraph, S—Sentences, T—Time-based.

Table A4. Extended results table of the variable encodings on SENTENCES experiments.

Trained/Tested on		PCM/GSM-FR	PCM/G.723.1	GSM-FR/PCM	G.723.1/PCM
i-vector	Spec.	78.2%	87.9%	85.6%	85.6%
	Sen.	94.7%	92.1%	92.6%	89.7%
	Prec.	81.3%	88.4%	86.5%	86.2%
	Acc.	86.5%	90.0%	89.1%	87.6%
	F1	87.5%	90.2%	89.5%	87.9%
x-vector	Spec.	82.4%	76.5%	72.9%	78.8%
	Sen.	78.8%	76.5%	78.8%	71.8%
	Prec.	81.7%	76.5%	74.4%	77.2%
	Acc.	80.6%	76.5%	75.9%	75.3%
	F1	80.2%	76.5%	76.6%	74.4%
LSTM	Spec.	81.7%	74.4%	83.7%	82.2%
	Sen.	82.4%	86.4%	78.5%	76.8%
	Prec.	81.5%	69.4%	85.0%	83.8%
	Acc.	82.1%	79.3%	80.9%	79.3%
	F1	82.0%	77.0%	81.6%	80.2%

Continued

SVM-linear	Spec.	81.5%	70.6%	83.8%	90.6%
	Sen.	84.7%	90.3%	83.5%	70.0%
	Prec.	82.1%	75.4%	83.8%	88.1%
	Acc.	83.1%	80.4%	83.7%	80.3%
	F1	83.4%	82.2%	83.7%	78.0%
SVM-rbf	Spec.	82.1%	76.8%	86.8%	89.7%
	Sen.	88.2%	91.8%	80.3%	77.4%
	Prec.	83.1%	79.8%	85.8%	88.3%
	Acc.	85.1%	84.3%	83.5%	83.5%
	F1	85.6%	85.4%	83.0%	82.4%

In the second column, the abbreviations are: Spec.—Specificity, Sen.—Sensitivity, Prec.—Precision, Acc.—Accuracy, F1—F1-score. In the first row, the abbreviations are: P—Paragraph, S—Sentences, T—Time-based.

Table A5. Extended results table of the variable encodings on TIME-BASED experiments.

Trained/Tested on		PCM/GSM-FR	PCM/G.723.1	GSM-FR/PCM	G.723.1/PCM
i-vector	Spec.	80.2%	85.3%	85.7%	84.9%
	Sen.	92.3%	86.7%	89.3%	86.6%
	Prec.	86.5%	89.0%	89.6%	88.8%
	Acc.	87.2%	86.1%	87.8%	85.9%
	F1	89.3%	87.9%	89.4%	87.7%
x-vector	Spec.	83.5%	76.5%	75.3%	84.7%
	Sen.	76.5%	76.5%	76.5%	65.9%
	Prec.	82.3%	76.5%	75.6%	81.2%
	Acc.	80.0%	76.5%	75.9%	75.3%
	F1	79.3%	76.5%	76.0%	72.7%
LSTM	Spec.	84.1%	80.9%	83.9%	85.1%
	Sen.	77.0%	81.1%	80.6%	74.3%
	Prec.	78.4%	71.5%	77.2%	80.9%
	Acc.	81.0%	81.0%	82.6%	80.1%
	F1	77.7%	76.0%	78.8%	77.4%

Continued

SVM-linear	Spec.	69.7%	70.5%	79.2%	86.7%
	Sen.	92.8%	91.4%	82.6%	73.2%
	Prec.	80.8%	81.0%	84.5%	88.3%
	Acc.	83.1%	82.6%	81.2%	78.9%
	F1	86.4%	85.9%	83.5%	80.0%
SVM-rbf	Spec.	72.8%	71.0%	82.6%	85.3%
	Sen.	92.4%	91.2%	82.2%	77.9%
	Prec.	82.4%	81.2%	86.7%	87.9%
	Acc.	84.2%	82.7%	82.4%	81.0%
	F1	87.1%	85.9%	84.4%	82.6%

In the second column, the abbreviations are: Spec.—Specificity, Sen.—Sensitivity, Prec.—Precision, Acc.—Accuracy, F1—F1-score. In the first row, the abbreviations are: P—Paragraph, S—Sentences, T—Time-based.

Table A6. Extended results table of the majority voting experiments on SVMs.

	Experiment	no MV	MV	no MV	MV
	Testing On	**Time-based**	**Time-based**	**Sentences**	**Sentences**
rbf kernel	Spec.	78.4%	81.2%	85.3%	83.5%
	Sen.	89.1%	92.9%	87.1%	91.8%
	Prec.	85.0%	83.2%	85.5%	84.8%
	Acc.	84.6%	87.1%	86.2%	87.6%
	F1	87.0%	87.8%	86.3%	88.1%
linear kernel	Spec.	74.7%	81.2%	84.1%	85.9%
	Sen.	88.4%	92.9%	85.0%	89.4%
	Prec.	82.8%	83.2%	84.3%	86.4%
	Acc.	82.6%	87.1%	84.6%	87.7%
	F1	85.5%	87.8%	84.6%	87.9%

In the second column, the abbreviations are: Spec.—Specificity, Sen.—Sensitivity, Prec.—Precision, Acc.—Accuracy, F1—F1-score. In the first row, the MV stands for the majority voting abbreviation.