applied sciences

MDPI

Article

A Hybrid Hidden Markov Model for Pipeline Leakage Detection

Mingchi Zhang [1,†], Xuemin Chen [2,*,†] and Wei Li [1,†]

[1] Department of Computer Science, Texas Southern University, Houston, TX 77004, USA; mingchi.zhang@gmail.com (M.Z.); wei.li@tsu.edu (W.L.)
[2] Department of Engineering, Texas Southern University, Houston, TX 77004, USA
* Correspondence: xuemin.chen@tsu.edu
† Current address: Texas Southern University, Houston, TX 77004, USA.

Abstract: In this paper, a deep neural network hidden Markov model (DNN-HMM) is proposed to detect pipeline leakage location. A long pipeline is divided into several sections and the leakage occurs in different section that is defined as different state of hidden Markov model (HMM). The hybrid HMM, i.e., DNN-HMM, consists of a deep neural network (DNN) with multiple layers to exploit the non-linear data. The DNN is initialized by using a deep belief network (DBN). The DBN is a pre-trained model built by stacking top-down restricted Boltzmann machines (RBM) that compute the emission probabilities for the HMM instead of Gaussian mixture model (GMM). Two comparative studies based on different numbers of states using Gaussian mixture model-hidden Markov model (GMM-HMM) and DNN-HMM are performed. The accuracy of the testing performance between detected state sequence and actual state sequence is measured by micro F_1 score. The micro F_1 score approaches 0.94 for GMM-HMM method and it is close to 0.95 for DNN-HMM method when the pipeline is divided into three sections. In the experiment that divides the pipeline as five sections, the micro F_1 score for GMM-HMM is 0.69, while it approaches 0.96 with DNN-HMM method. The results demonstrate that the DNN-HMM can learn a better model of non-linear data and achieve better performance compared to GMM-HMM method.

Keywords: deep neural network; hidden Markove model; leakage detection; pipeline

Citation: Zhang, M.; Chen, X.; Li, W. A Hybrid Hidden Markov Model for Pipeline Leakage Detection. *Appl. Sci.* **2021**, *11*, 3138. https://doi.org/10.3390/app11073138

Academic Editor: Giuseppe Lacidogna

Received: 8 March 2021
Accepted: 30 March 2021
Published: 1 April 2021

1. Introduction

Damage detection has been widely studied especially in the pipeline to avoid enormous economic loss and environmental disasters [1]. Pipeline leakage detection is an essential component of risk management as it allows operators to respond in time to leaks and to prevent further escalation of incidents. In the last few decades, many pipeline leakage detection techniques have been used to monitor damages for the offshore oil and gas industry. Existing leakage detection techniques are acoustic emission [2], fiber optic sensing [3], negative pressure wave (NPW) detection [4], etc. The extracted signals from these techniques can be analyzed with statistical models to monitor the damages. For instance, statistical models, such as support vector machine (SVM) [5–7] and hidden Markov model (HMM) [8,9], have been adopted to facilitate and enhance the pipeline detection process utilizing the various extracted signals from the detection techniques. Liu et al. evaluated a deep neural network for spectrum recognition of underwater targets [10]. Sohaib et al. compared detection performance between statistic models on the boiler tube [11]. However, these studies only focus on classification without considering the sequential changes of damage states. Furthermore, to exploit the feature information in signals, Gaussian mixture model-hidden Markov model (GMM-HMM) has been implemented to improve the pipeline damage detection performance [12,13].

In our previous study, GMM-HMM was applied for pipeline leakage and crack depth detection [14]. Each hidden state has a probability distribution over the possible leakage states where the probability distribution matrix is initialized by a Gaussian mixture model

(GMM). An iterative training process through the Baum–Welch algorithm is applied to get the optimized parameters of the HMM. The posterior probabilities of leakage states obtained by HMM is used to obtain a probabilistic evaluation of the pipeline damage. The proposed GMM-HMM can recognize the crack depth by using the guided wave and leakage location of the pipeline by using the negative pressure wave. Signals were collected by Lead Zirconate Titanate (PZT) transducers. Different crack depths and different leakage locations of pipeline were defined as different states in the hidden Markov model. That work successfully answered our research questions, i.e., whether the pipeline has a leak, where the leakage location is and how deep the crack is. In addition, the GMM-HMM method [14] has the ability to detect the sequential changes of states. However, GMM-HMM becomes less efficient for pipeline leakage detection due to the demands of a massive amount of data to identify parameters of Gaussian mixtures. GMM typically has a large number of Gaussians when there are many hidden states. The GMM with independently parameterized means from such states may result in those Gaussians being highly localized and thus may result in such models only performing local generalization. This situation will become worse when changing environmental factors are considered. Therefore, in the real-world application, the changing environment and time-varying operational conditions make the reliability of pipeline leakage detection facing the challenge.

To overcome this predicament, one of the techniques is to replace the GMM with reliable models that can tackle with a massive amount of data and achieve higher accuracy. With the surging of deep learning, neural networks can model multiple events and learn richer representations that have the potential to learn better models of nonlinear data [15–17]. With multiple layers, deep neural networks (DNNs) [18,19] perform well on decision boundary and feature engineering problems by using a massive amount of data [20]. In recent years, a deep neural network hidden Markov model (DNN-HMM) has been proposed as a novel hybrid architecture and has been widely used on acoustic learning [21–23]. DNN computes the emission probabilities of states for the HMM, which offers strong ability of feature learning and provides a better recognition result [24]. In recent research, Qiu et al. presented an early-warning model of equipment chain in gas pipeline based on DNN-HMM which demonstrated preferable generalization accuracy [25]. Schröder et al. conducted comparison study between GMM-HMM and DNN-HMM for acoustic event detection and demonstrated that the performance may varied by using different features [26].

In this study, the DNN-HMM hybrid model is proposed to detect pipeline leakage locations as different states from lead zirconate titanate (PZT) transducer signals generated by negative pressure wave. In the proposed DNN-HMM hybrid model, DNN consists of the unsupervised deep belief network (DBN) that computes the emission probabilities of leakage states for the HMM instead of GMM. First, the DBN pre-training is used to make sure that training would be effective. The DBN is built by stacking top-down restricted Boltzmann machines (RBMs). The RBM is bipartite connectivity structured and has an unobserved subset of the variables. The cyclic process of serve inferred the result of one RBM as training data for another RBM generated a multilayer feature detector. Based on the pre-trained network, the DNN will outperform the random initialization for more complex statistical structure extracted from the PZT signal. Then, the posterior probabilities of leakage state as the output of the DNN will serve as the input parameter for HMM. Differing from Tejedor et al. [13], who implemented the GMM-HMM based pattern classification to monitor the pipeline integrity, the DNN-HMM is implemented in our study to detect the pipeline leakage location. Thus far, there is only one work, presented by Qiu et al. [25], that applied DNN-HMM for classifying the generalized damage causation in equipment chain. Different from the existing work, we extract one time domain index and one frequency domain index from noisy negative pressure wave signals collected by PZT transducers when leakage occurs as observations for the proposed hybrid HMM method. To illustrate the effectiveness of proposed method, two groups of experiments that contain

different states are conducted. The comparison results of DMM-HMM and GMM-HMM are also presented in the paper.

The rest of paper is organized as follows. The hybrid HMM method is presented in Section 2. The experimental results and DNN-HMM leakage location detection results are presented in Section 3. The conclusions are drawn and future work is discussed in Section 4.

2. Hybrid HMM Method

In this section, we propose a deep neural network hidden Markov model for pipeline leakage location detection. In the proposed hybrid HMM method, DNN computes the emission probabilities for the HMM instead of Gaussian mixtures. In the following subsections, the main components of the proposed method, including the typical HMM, DNN-DBN pre-training, hybrid HMM, are presented.

2.1. Hidden Markov Model

The hidden Markov model is a probabilistic graphical model in which the unobservable ("hidden") system state sequence is modeled by Markov chain, and hidden states can be indirectly observed by observation states through some probabilistic distribution. HMM was first proposed by Baum and Petrie in 1966 [27]. HMM and its variants have been widely used in different application areas [28].

A typical HMM model [29] can be defined by

$$\lambda = \{\pi, A, B\} \quad , \tag{1}$$

where π is the initial probability distribution matrix, A is the state transition probability matrix and B is the emission probability matrix or observation probability distribution matrix. π, A and B are row stochastic matrices.

In this paper, a long pipeline is divided into several sections and the leakage occurs in a different section that is defined as a different state of the hidden Markov model. The collected stress waves are used to extract the damage indices which serve as observation data. A HMM with three states and three observation states is shown in Figure 1.

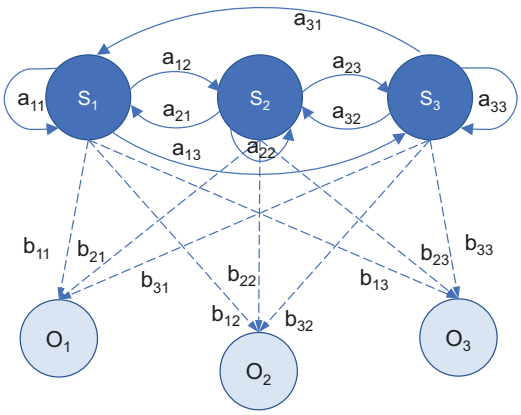

Figure 1. A hidden Markov model with three states and three observation states.

2.2. DNN-DBN Pre-Training

DNN is a feed-forward, conventional artificial neural network with a multi-layer of hidden units, regularly initialized by using the DBN pre-training algorithm. The estimates of the posterior probabilities computed by the neural network are divided by the prior state probabilities, resulting in scaled likelihoods, which are used as emission probabilities in the HMM. When it is trained on a dataset without supervision, a DBN can learn the probabilistic reconstruction of its inputs. The layers then act as feature detectors. After this learning step, a DBN can be further trained with supervision to perform classification [30].

For each hidden unit, the sigmoid function is typically used to map total input from current layer, as shown in Equation (2) [31],

$$\begin{aligned} y_j = logistic(x_j) = \frac{1}{1+e^{-x_j}} \quad , \\ x_j = b_j + \sum_i y_i w_{ij} \quad , \end{aligned} \tag{2}$$

where j is the hidden unit index from range of 0 to N, N is a finite positive integer, x_j is the current layer state, y_j represents the current layer output, b_j is the bias of current unit, i is an index over units in the layer below, and $w_{i,j}$ is the weight between two successive hidden units. For multiclass classification, the softmax function is applied to convert the DNN output into a class probability. The cost function is defined as cross-entropy between the target probabilities and the output probabilities from softmax function, as shown in Equation (3):

$$C = -\sum_{j=0}^{N} d_j \log P_j \quad , \tag{3}$$

where C denotes the cost function, d_j represents the target probabilities and P_j is the probabilities of outputs. Thus, the target probabilities of the HMM states are the learned information provided to train the DNN. One of the benefits of using a DNN is that it can efficiently computes derivatives before updating the weights in proportion to the gradient by dividend large datasets into random minibatches. This stochastic gradient descent method can be further improved by using a momentum coefficient [21].

DBN provides a new way to train deep generative models by using layer-wise pre-training of RBMs. In this way, the pre-training process will provide proper initial weights for the DBN. A DBN that consists of visible layers and hidden layers is shown in Figure 2.

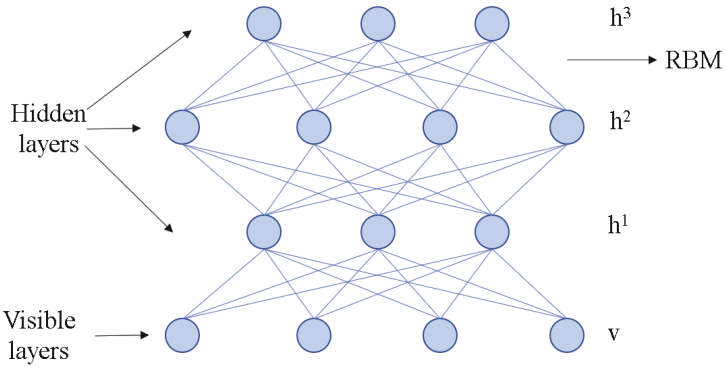

Figure 2. DBN structure.

Two successive hidden layers in DBN form an RBM. The DBN is composed by stacked modules of RBMs. Low-dimensional features are extracted from input data by pre-training of the DBN without losing much significant information. Each RBM in the DBN is a bipartite graph where there is no connection between each hidden unit on the same layer. The joint probability of an RBM is defined as

$$P_{h,v}(RBM) = \frac{1}{Z_{h,v}} e^{v^t W h + v^t b + a^T h} \quad , \tag{4}$$

where h defined as hidden vectors for the Bernoulli–Bernoulli RBM [32] applied to binary v with a second bias vector b, bias vector a, weight matrices W and normalization term $Z_{h,v}$.

$$P_{h,v}(Gaussian - Bernoulli\ RBM) = \frac{1}{Z_{h,v}} e^{v^t W h + (v-b)^t (v-b) + a^T h} \quad , \tag{5}$$

where Gaussian–Bernoulli RBM (GRBM) [32] is applied to continuous v. In both cases, the conditional probability $P_{h,v}(h, v)$ has the same form as that in the DNN [20]. The RBM parameters can be efficiently trained in an unsupervised fashion by maximizing the likelihood over training samples with the approximate contrastive divergence algorithm [21].

2.3. Integrating DNN with HMM

The emission probability for the HMM is represented as $P(x|a)$, while the output of DBN-DNN is $P(a|x)$. The following formula calculates the transformation.

$$P(x|a) = \frac{P(x,a)}{P(a)} = \frac{P(a|x)P(x)}{P(a)} \quad , \tag{6}$$

where classes x correspond to HMM states, with observation vectors a, and $P(a)$ count from training data.

2.4. Architecture of DNN-HMM

The DNN is built based on the DBN structure. The training process of the DBN is shown in Figure 3.

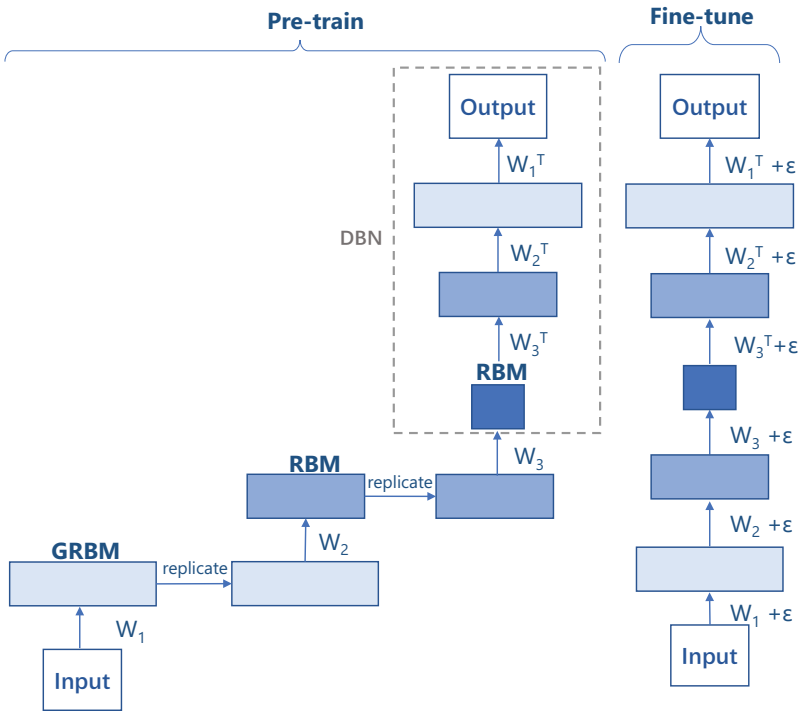

Figure 3. DBN Training Process.

Typically, the training process contains two phases: greedy pre-training and fine tuning. First, pre-training is applied to obtain the proper initial weights (e.g., W_1, W_2 and W_3) of the DBN. In the pre-training phase, the input, i.e., damage indices extracted from PZT sensor, is modeled and trained by a GRBM. After training the RBM using the training set, the inferred states of the hidden units can be used as data for training another RBM that learns to model the significant dependencies between the hidden units of the first RBM. This can be repeated as many times as desired to produce many layers of non-linear feature detectors that represent progressively more complex statistical structures in the PZT sensor signals. The complex statistical structure also represents the complex distributions of the pipeline leakage signals. Stacking RBMs by replacing the connections of lower level RBMs top-down (W_3 to W_3^T, W_2 to W_2^T and W_1 to W_1^T), an unsupervised training procedure DBN is made, as shown in the Figure 3. In this way, the DBN is able to learn the

complex distribution of the pipeline leakage signals. Then, fine-tuning is used to optimized the initial weights with more pipeline leakage signals. Therefore, the initial weights are updated with small errors (ϵ) to get more accurate representation of the distribution.

After the fine-tuning process of the DBN, the optimized proper weights (W_1', W_2' and W_3') can be used to construct the DBN-DNN by adding a softmax output layer, as shown in Figure 4. This softmax output layer contains one unit for each possible state of the HMM. In this way, the posterior probability output from DBN-DNN for each state can be converted into the emission probability as the input parameter of the HMM. The initial state distribution π, transition probability matrix A and emission probability matrix B of the HMM are updated by training with iterations of expectation and maximization steps to obtain the trained model.

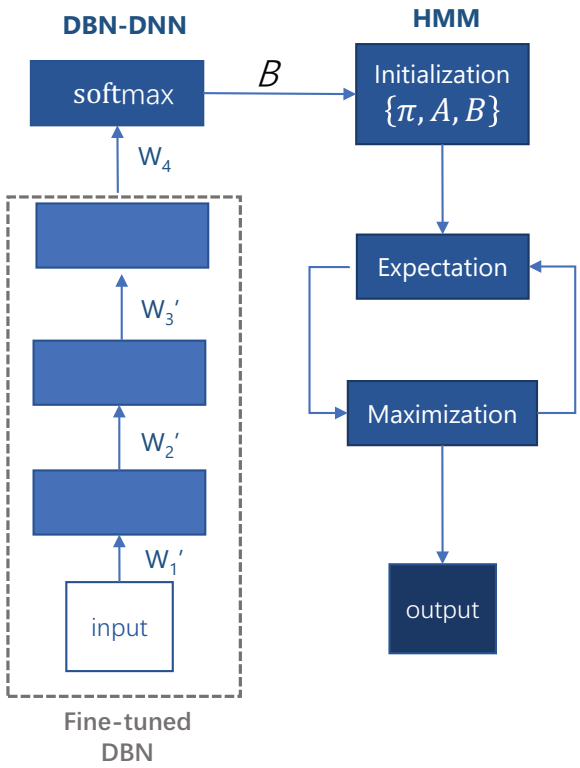

Figure 4. DBN-DNN-HMM architecture.

3. Pipeline Leakage Detection

In this section, we evaluate the proposed method by comparing it with GMM-HMM method. In the following subsections, we provide the experimental settings, damage indices extraction, DBN-DNN training result and evaluation metrics. The accuracy comparisons of hybrid HMM and GMM-HMM methods for three states and five states are presented.

3.1. Setup of Experiment

The purpose of experiment is to detect pipeline leakage utilizing PZT transducers. The negative pressure wave is generated by leakage in the pipeline and propagated along the pipeline from the leakage point to both ends. The experimental pipeline was built at the University of Houston, as shown in Figure 5.

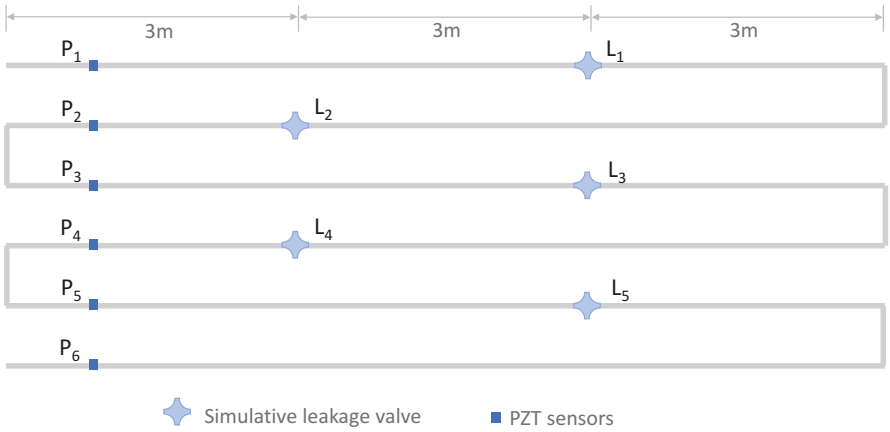

Figure 5. Settlement of the pipeline and sensors.

The experimental pipeline consists of a series of Plain-End PVC pipe sections connected together to form a pipeline with a total length of 55.78 m. Six PZT transducers (P_1 to P_6, size is 15 mm $\times$ 10 mm) are directly mounted on the pipeline by using epoxy to detect negative pressure wave signal. A NI PXI$-$5105 digitizer is used as a data acquisition system. The digitizer is triggered by the voltage signal of PZT No. 1 with the trigger level at -0.02 V and all the signals from six PZT sensors are recorded simultaneously at a sampling rate of 100 KS/s. The experiment was presented in a published paper using the latency of signal to locate leakage [4]. In [4], the PZT sensors were used to detect the arrival time of NPW, and then the arrival time was used to calculate the exact location of leakage. We use the same experimental setting and data to validate the proposed DNN-HMM model which is for detecting leakage section and sequential changes of damage states. The hybrid model is implemented based on the hmmlearn [33] and deep-belief-network Python package [34], ran on a desktop with Windows 10 64-bit and Intel core i7-8700 CPU. The seqHMM R package is used for Gaussian mixture model component number K estimation.

Different leakage locations are chosen as different states in the HMM. Two damage indices, i.e., one time domain damage index and one frequency domain index, as we used in [14], are extracted from the original signals. The damage indices are adopted to indicate the signal variations and serve as observations of the HMM. The first damage index (DI_1) is a time-domain damage index, defined as:

$$DI_1 = 1 - \sqrt{\frac{\sum_{t=0}^{t'}(s_1(t) - \bar{s}_1)(s_2(t) - \bar{s}_2)}{\sum_{t=0}^{t'}(s_1(t) - \bar{s}_1)^2 \sum_{t=0}^{t'}(s_2(t) - \bar{s}_2)^2}} \quad , \tag{7}$$

where $s_1(t)$ is the baseline waveform and $s_2(t)$ is the comparison waveform at time t. $\bar{s}_1$ and $\bar{s}_2$ are the average values of $s_1(t)$ and $s_2(t)$.

The second one is a frequency-domain damage index DI_2, the amplitude of peak frequency, defined as:

$$DI_2 = \max_{0 \le K \le L-1} (|X(K)|) \quad , \tag{8}$$

where $X(K) = \sum_0^{N-1} X(n) e^{-i2\pi/L}$, $i = \sqrt{-1}$. K is frequency corresponding to the selected frequency spectrum window and L is length of signal. The leakage signal collected by the first sensor P_1 serves as the original data for damage indices extraction.

3.2. Setup of DBN-DNN

The DBN-DNN model replaces the GMM, offering the posterior probabilities to the HMM as the input emission probability. Ten layers of the RBM and 200 layers of the DNN are applied in the training process. The learning rate of RBM is 0.05, and the DNN learning rate is set to 0.1. The maximum number of iterations of DNN is 200, with dropout rate of

0.2. The rectified linear unit (ReLu) activation function is applied in the training process of DBN-DNN. ReLu is a non-linear activation function that is used in multi-layer neural networks or deep neural networks.

3.3. Performance of DBN-DNN

After 100 epochs, the cross-entropy of the DBN-DNN is obtained by using Equation (3), which approaches 0.08. The output probabilities are very close to the target probabilities. The performance of DBN-DNN is shown in Figure 6.

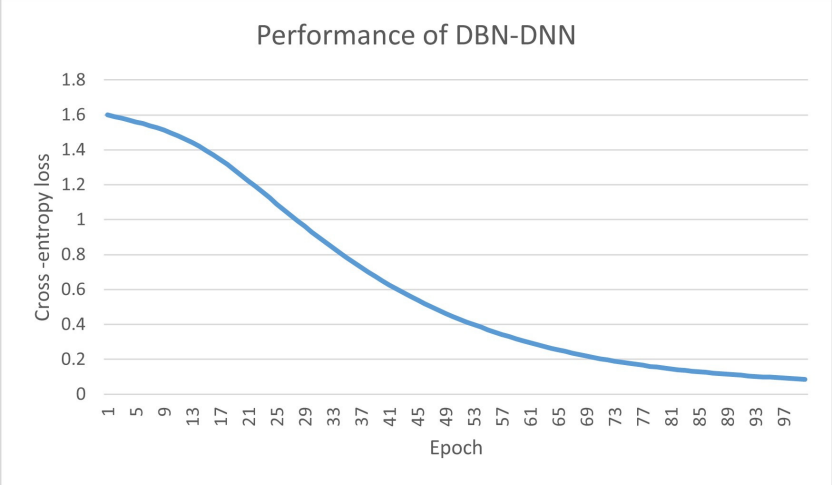

Figure 6. Performance of DBN-DNN.

3.4. Performance Evaluation

Normally, F_1 score measurement is applied for binary classification, which is calculated from precision and recall [35]:

$$F_1\ score = \frac{2 \times TP}{2 \times TP + FP + FN} \quad , \tag{9}$$

where TP is the number of true positive, FP represents the number of the false-positive and FN is the number of false negative.

F_1 score has been used to measure the multi-class classification also [36]. The F_1 score can be interpreted as a weighted harmonic mean of precision and recall, where an F_1 score reaches its best value at 1 and the worst score at 0. There are two types of F_1 score, micro F_1 score and macro F_1 score. Micro F_1 score is calculated by measuring the F_1 score with the aggregated contribution of all classes. Macro F_1 is calculated by averaging the precision and recall of all instances and dominated by the performance of common categories [37]. Thus, micro F_1 score is adopted in the performance measurement in this study. Micro F_1 score is defined in the following equation:

$$Micro\ F_1\ score = 2 \times \frac{(P_1 + P_2 + \ldots + P_{m-1} + P_m) \times (R_1 + R_2 + \ldots + R_{m-1} + R_m)}{(P_1 + P_2 + \ldots + P_{m-1} + P_m) + (R_1 + R_2 + \ldots + R_{m-1} + R_m)} \tag{10}$$

where P_i represents precision and R_i represents recall, $i = 1, \ldots, m$, and m is the class number of classifications.

3.5. Leakage Detection with Three States

In our previews work [14], three different leakage locations were chosen as three states for the GMM-HMM method. To compare the performance of DNN-HMM with GMM-HMM, the same state setting and initial parameters were applied in the two models. The leakage signals were collected by sensor P_1. As shown in Figure 7, leakage at Section 1 (from P_1 to L_1 in Figure 5) of the pipeline as State 1, leakage at Section 2 (from L_1 to L_2) of

the pipeline as State 2, leakage at Section 3 (from L_2 to L_3) of the pipeline as State 3 are chosen as states in ergodic HMM model, which allowed transitions from any state to any other state.

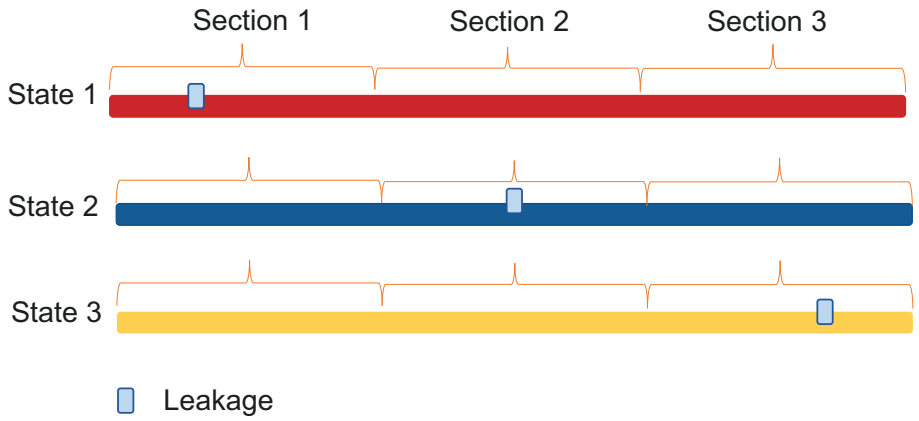

Figure 7. Schematic diagram of three pipeline states.

100 groups, with 200,000 samples per group, of leakage signals were collected. In this study, 70 groups are used for training and the rest are used for testing. To reduce the computational volume, data for this experiment are cropped from the original data by 90,000 data points to each state. Two damage indices are obtained by using Equations (7) and (8) with 45 data points for each state in HMM.

The micro F_1 score is calculated to measure the performance. The micro F_1 score of GMM-HMM is 0.94 and the micro F_1 score of DNN-HMM approaches 0.95. Several tests are made to compare the performance of DNN-HMM and GMM-HMM, where the performance of DNN-HMM is little better than GMM-HMM in almost every trial. One of the experiment results is shown in Figure 8.

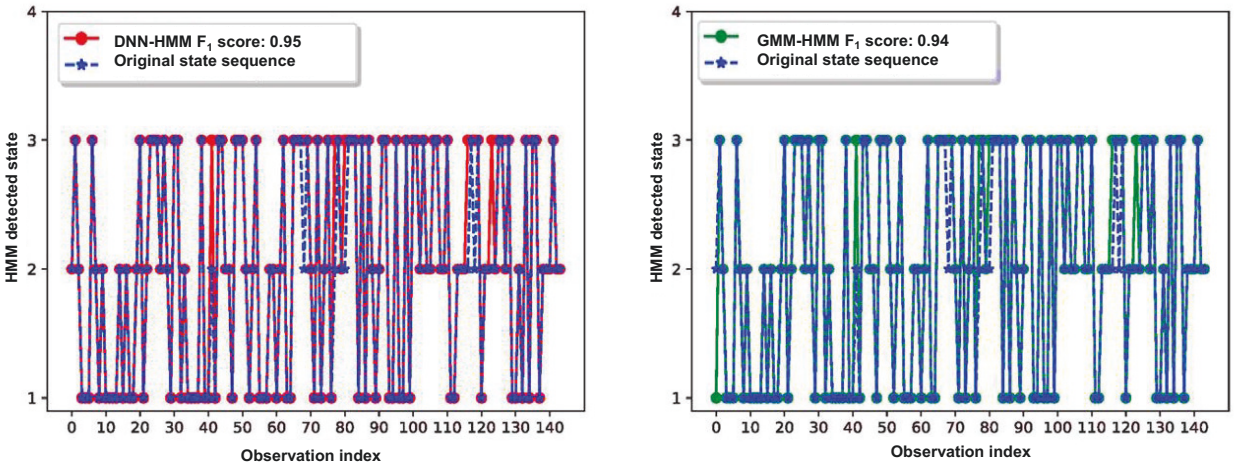

Figure 8. Comparison between GMM-HMM and DNN-HMM with three states.

3.6. Leakage Detection with Five States

In this experiment, five leakages at different section of pipeline are chosen as the five states in ergodic HMM model (Figure 9). Leakage signals are collected by sensor P_1. As shown in Figure 7, leakage at Section 1 (from P_1 to L_1 in Figure 5) of the pipeline as State 1, leakage at Section 2 (from L_1 to L_2) of the pipeline as State 2, leakage at Section 3 (from L_2

to L_3) of the pipeline as State 3, leakage at Section 4 (from L_3 to L_4) of the pipeline as State 4 and leakage at Section 5 (from L_4 to L_5) of the pipeline as State 5 are considered.

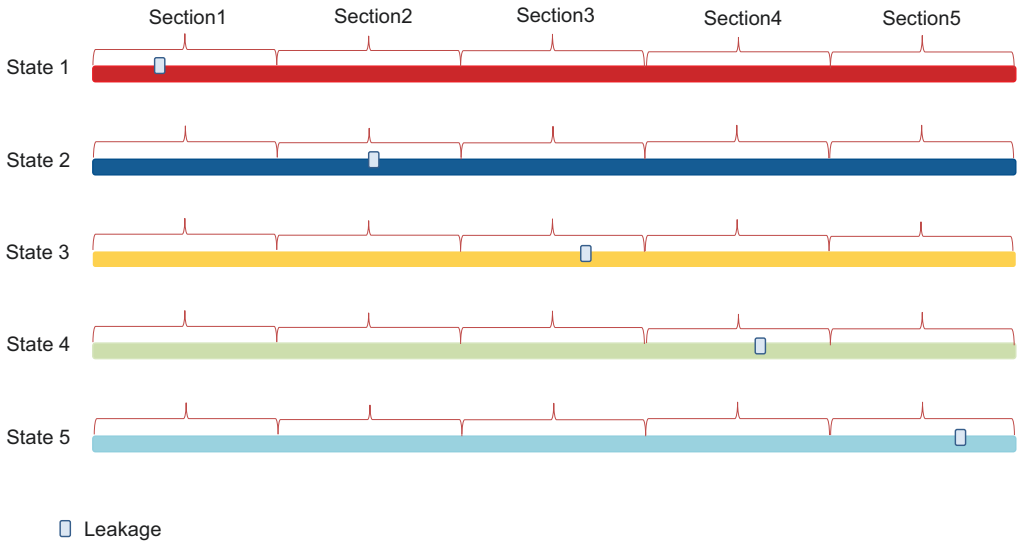

Figure 9. Schematic diagram of 5 states pipeline statues.

For each location, the leakage experiment is repeated for 20 times. Each experiment generated 100 data points which contain the damage indices (DI_1 and DI_2). In total, 600 data points are extracted for the hybrid model, as shown in Figure 10. Among all these extracted damage indices, 80% of the data points are used for training, while the rest are used for testing the performance of hybrid model.

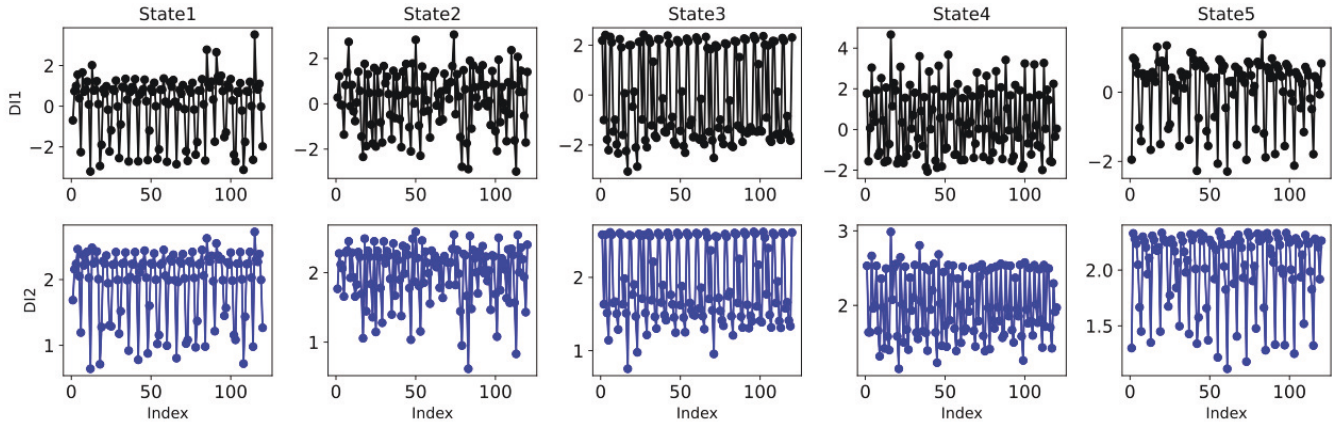

Figure 10. Damage Indices.

The testing performance between detected state sequence and actual state sequence of DNN-HMM is calculated by micro F_1 score, which approaches 0.96, while the micro F_1 score of GMM-HMM is about 0.69, as shown in Figure 11. Compared with the performance of GMM-HMM and DNN-HMM model initialize with three states, the increase of states reduces the accuracy of GMM-HMM model. However, in both experiments, DNN-HMM performed better than GMM-HMM.

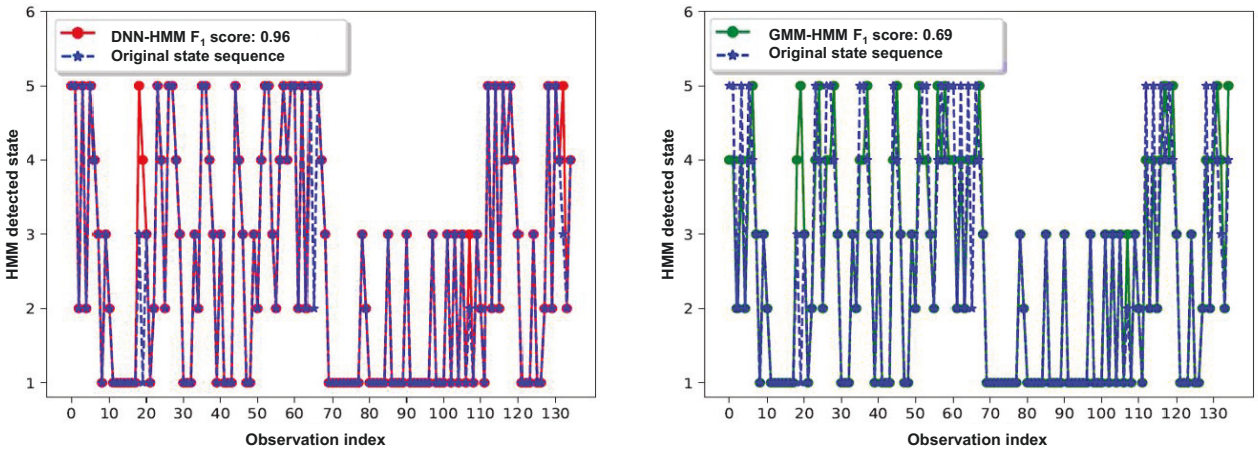

Figure 11. Comparison between GMM-HMM ant DNN-HMM with five states.

4. Conclusions

In this paper, a DNN-HMM hybrid model was proposed to detect pipeline leakage location. DNN computes the emission probabilities for the HMM instead of Gaussian mixture model. This hybrid model showed the feasibility of converting leakage state posteriors to the emission probabilities by training a DNN which uses the damage indices as the training set. DNN is more efficient for modeling leakage features. The DNN is a pre-trained model built by stacking top-down RBM that computes the emission probabilities for the HMM. Two comparative tests based on different numbers of states using GMM-HMM and DNN-HMM were studied. The results demonstrate that the DNN-HMM can learn better models of data and achieve better performance compared to GMM-HMM. The micro F_1 score approaches 0.94 for three states, and it approaches 0.96 for five states when the hybrid HMM was applied.

In this paper, two damage indices, i.e., one time domain damage index and one frequency domain index, were used to extract features from the negative pressure waves collected by PZT transducers. In future work, other damage indices or even without using damage indices for pipeline leakage detection will be explored.

Author Contributions: Conceptualization, M.Z., X.C. and W.L.; software, M.Z.; validation, M.Z., X.C. and W.L.; writing—original draft preparation, M.Z.; writing—review and editing, X.C. and W.L.; visualization, M.Z.; project administration, X.C.; and funding acquisition, X.C. All authors have read and agreed to the published version of the manuscript.

Funding: This material is based upon work supported by the National Science Foundation under Grants No. 1801811.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Thanks to Smart Materials and Structures Laboratory at the University of Houston for providing the pipeline leakage experimental data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jin, H.; Zhang, L.; Liang, W.; Ding, Q. Integrated leakage detection and localization model for gas pipelines based on the acoustic wave method. *J. Loss Prev. Process Ind.* **2014**, *27*, 74–88.
2. Liang, W.; Zhang, L.; Xu, Q.; Yan, C. Gas pipeline leakage detection based on acoustic technology. *Eng. Fail. Anal.* **2013**, *31*, 1–7.

3. Zhou, Y.; Jin, S.J.; Zhang, Y.C.; Sun, L.y. Study on the distributed optical fiber sensing technology for pipeline leakage detection. *J. Optoelectron. Laser* **2005**, *16*, 935.
4. Zhu, J.; Ren, L.; Ho, S.C.; Jia, Z.; Song, G. Gas pipeline leakage detection based on PZT sensors. *Smart Mater. Struct.* **2017**, *26*, 025022.
5. Qu, Z.; Feng, H.; Zeng, Z.; Zhuge, J.; Jin, S. A SVM-based pipeline leakage detection and pre-warning system. *Measurement* **2010**, *43*, 513–519.
6. Li, Q.; Du, X.; Zhang, H.; Li, M.; Ba, W. Liquid pipeline leakage detection based on moving windows LS-SVM algorithm. In Proceedings of the 2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC), Nanjing, China, 18–20 May 2018; pp. 701–705, doi:10.1109/yac.2018.8406462.
7. Kang, J.; Park, Y.J.; Lee, J.; Wang, S.H.; Eom, D.S. Novel leakage detection by ensemble CNN-SVM and graph-based localization in water distribution systems. *IEEE Trans. Ind. Electron.* **2017**, *65*, 4279–4289.
8. Ai, C.; Zhao, H.; Ma, R.; Dong, X. Pipeline damage and leak detection based on sound spectrum LPCC and HMM. In Proceeding sof the Sixth International Conference on Intelligent Systems Design and Applications, Jian, China, 16–18 October 2006; Volume 1, pp. 829–833.
9. Ai, C.; Sun, X.; Zhao, H.; Ma, R.; Dong, X. Pipeline damage and leak sound recognition based on HMM. In Proceedings of the 2008 7th World Congress on Intelligent Control and Automation, Chongqing, China, 25–27 June 2008; pp. 1940–1944.
10. Liu, D.; Zhao, X.; Cao, W.; Wang, W.; Lu, Y. Design and Performance Evaluation of a Deep Neural Network for Spectrum Recognition of Underwater Targets. *Comput. Intell. Neurosci.* **2020**, *2020*, doi:10.1155/2020/8848507.
11. Sohaib, M.; Kim, J.M. Data driven leakage detection and classification of a boiler tube. *Appl. Sci.* **2019**, *9*, 2450.
12. Tejedor, J.; Macias-Guarasa, J.; Martins, H.; Martin-Lopez, S.; Gonzalez-Herraez, M. A Gaussian Mixture Model-Hidden Markov Model (GMM-HMM)-based fiber optic surveillance system for pipeline integrity threat detection. In *Optical Fiber Sensors*; Optical Society of America: Washington, DC, USA, 2018; p. WF36.
13. Tejedor, J.; Macias-Guarasa, J.; Martins, H.F.; Martin-Lopez, S.; Gonzalez-Herraez, M. A contextual GMM-HMM smart fiber optic surveillance system for pipeline integrity threat detection. *J. Light. Technol.* **2019**, *37*, 4514–4522.
14. Zhang, M.; Chen, X.; Li, W. Hidden Markov Models for Pipeline Damage Detection Using Piezoelectric Transducers. *arXiv* **2020**, arXiv:2009.14589.
15. Lee, C.; Landgrebe, D.A. Decision boundary feature extraction for neural networks. *IEEE Trans. Neural Netw.* **1997**, *8*, 75–83.
16. Lan, Q.; Zheng, J.; Chen, J. Predicting MRI RF Exposure for Complex-shaped Medical Implants Using Artificial Neural Network. In Proceedings of the 2019 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting, Atlanta, GA, USA, 7–12 July 2019; pp. 1861–1862.
17. Zheng, J.; Lan, Q.; Zhang, X.; Kainz, W.; Chen, J. Prediction of MRI RF exposure for implantable plate devices using artificial neural network. *IEEE Trans. Electromagn. Compat.* **2019**, *62*, 673–681.
18. Qian, Y.; Fan, Y.; Hu, W.; Soong, F.K. On the training aspects of deep neural network (DNN) for parametric TTS synthesis. In Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 4–9 May 2014; pp. 3829–3833.
19. Woo, S.; Lee, C. Feature extraction for deep neural networks based on decision boundaries. In *Pattern Recognition and Tracking XXVIII*; International Society for Optics and Photonics: Bellingham, WA, USA, 2017; Volume 10203, p. 1020306.
20. Seide, F.; Li, G.; Chen, X.; Yu, D. Feature engineering in context-dependent deep neural networks for conversational speech transcription. In Proceedings of the 2011 IEEE Workshop on Automatic Speech Recognition & Understanding, Waikoloa, HI, USA, 11–15 December 2011; pp. 24–29.
21. Hinton, G.; Deng, L.; Yu, D.; Dahl, G.E.; Mohamed, A.R.; Jaitly, N.; Senior, A.; Vanhoucke, V.; Nguyen, P.; Sainath, T.N.; et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Process. Mag.* **2012**, *29*, 82–97.
22. Zen, H.; Senior, A. Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis. In Proceedings of the 2014 IEEE international conference on acoustics, speech and signal processing (ICASSP), Florence, Italy, 4–9 May 2014; pp. 3844–3848.
23. Chan, W.; Lane, I. Deep recurrent neural networks for acoustic modelling. *arXiv* **2015**, arXiv:1504.01482.
24. Dahl, G.E.; Yu, D.; Deng, L.; Acero, A. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Trans. Audio Speech Lang. Process.* **2011**, *20*, 30–42.
25. Qiu, J.; Liang, W.; Zhang, L.; Yu, X.; Zhang, M. The early-warning model of equipment chain in gas pipeline based on DNN-HMM. *J. Nat. Gas Sci. Eng.* **2015**, *27*, 1710–1722.
26. Schröder, J.; Anemüller, J.; Goetze, S. Performance comparison of GMM, HMM and DNN based approaches for acoustic event detection within task 3 of the DCASE 2016 challenge. In Proceedings of the Detection and Classification of Acoustic Scenes and Events, Budapest, Hungary, 3 September 2016; pp. 80–84.
27. Baum, L.E.; Petrie, T. Statistical inference for probabilistic functions of finite state Markov chains. *Ann. Math. Stat.* **1966**, *37*, 1554–1563.
28. Mor, B.; Garhwal, S.; Kumar, A. A systematic review of hidden Markov models and their applications. *Arch. Comput. Methods Eng.* **2020**, 1–20. doi:10.1007/s11831-020-09422-4.
29. Rabiner, L.R. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* **1989**, *77*, 257–286.

30. Wang, D.; Shang, Y. Modeling physiological data with deep belief networks. *Int. J. Inf. Educ. Technol.* **2013**, *3*, 505.
31. Hinton, G.E. A practical guide to training restricted Boltzmann machines. In *Neural Networks: Tricks of the Trade*; Springer: Berli/Heidelberg, Germany, 2012; pp. 599–619.
32. Yamashita, T.; Tanaka, M.; Yoshida, E.; Yamauchi, Y.; Fujiyoshii, H. To be Bernoulli or to be Gaussian, for a restricted Boltzmann machine. In Proceedings of the 22nd International Conference on Pattern Recognition, Stockholm, Sweden, 24–28 August 2014; pp. 1520–1525.
33. Lebedev, S. hmmlearn/hmmlearn: Hidden Markov Models in Python, with scikit-learn like API. 2016. Available online: https://github.com/hmmlearn/hmmlearn (accessed on 30 March 2021).
34. Albertbup. A Python Implementation of Deep Belief Networks Built upon NumPy and TensorFlow with Scikit-Learn Compatibility. 2017. Available online: https://github.com/albertbup/deep-belief-network (accessed on 30 March 2021).
35. Chicco, D.; Jurman, G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genom.* **2020**, *21*, 6.
36. Fujino, A.; Isozaki, H.; Suzuki, J. Multi-label text categorization with model combination based on F1-score maximization. In Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II, Hyderabad, India, 7–12 January 2008.
37. Liu, C.; Wang, W.; Wang, M.; Lv, F.; Konan, M. An efficient instance selection algorithm to reconstruct training set for support vector machine. *Knowl. Based Syst.* **2017**, *116*, 58–73.