 https://doi.org/10.1051/e3sconf/202122901003

From Auto-encoders to Capsule Networks: A Survey

Omaima El Alaoui-Elfels, Taoufiq Gadi
University Hassan First, Faculty of Science and Technology of Settat, Computing, Imaging and Modeling of Complex Systems Laboratory, Morocco
elalaoui-elfels.fst@uhp.ac.ma, gtaoufiq@yahoo.fr

Keywords: Convolutional Neural Networks, Auto-encoders, Capsule Networks, Routing by Agreement Between Capsules, EM Routing, Stacked Capsule Network, Deep Learning.

Abstract: Convolutional Neural Networks are a very powerful Deep Learning structure used in image processing, object classification and segmentation. They are very robust in extracting features from data and largely used in several domains. Nonetheless, they require a large number of training datasets and relations between features get lost in the Max-pooling step, which can lead to a wrong classification. Capsule Networks(CapsNets) were introduced to overcome these limitations by extracting features and their pose using capsules instead of neurons. This technique shows an impressive performance in one-dimensional, two-dimensional and three-dimensional datasets as well as in sparse datasets. In this paper, we present an initial understanding of CapsNets, their concept, structure and learning algorithm. We introduce the progress made by CapsNets from their introduction in 2011 until 2020. We compare different CapsNets series architectures to demonstrate strengths and challenges. Finally, we quote different implementations of Capsule Networks and show their robustness in a variety of domains. This survey provides the state-of-the-artof Capsule Networks and allows other researchers to get a clear view of this new field. Besides, we discuss the open issues and the promising directions of future research, which may lead to a new generation of CapsNets.

1 INTRODUCTION

Imitating the human brain used to be a dream for scientists until the creation of Artificial Neural Networks (ANNs). ANNs are the artificial version of Biological Neural Networks that constitute our nervous system. Simulating human brain ability in object classification was the goal of Convolutional Neural Networks (CNNs). This kind of neural networks shows high performance in object classification and image processing. CNNs extract the most significant features from images and use them for classification. However, CNNs are not able to detect object deformation and relationships among object entities. These limitations may lead to incorrect classification, hence influencing the model performance negatively.

Capsule Networks have been introduced to adjust CNNs and overcome their shortcomings. These networks are a combination of Auto-encoders and capsules. Auto-encoders (AE) are simple neural networks consisting of an encoder, latent space representation and decoder. The encoder compresses the input to latent space representation, then the decoder reconstructs the input based on this representation only. The network is trained by updating weights using backpropagation with a gradient optimizer. This kind of network is used for data denoising, dimensionality reduction and generative model. They were widely developed to extract more features while keeping the capacity of generalization, by Denoising AE (Vincent et al., 2008), Sparse AE (H. Lee et al., 2008), Variational AE (Pu et al., 2016) and Transforming AE(Hinton et al., 2011).

The introduction of Capsule Networks was in 2011. They were presented as Transforming AE by (Hinton et al., 2011) who noticed that Convolutional Neural Networks are misguided in what they are trying to achieve. CNNs lose meaningful information like object entities' poses and relationships between features in the Max-pooling layer. Transforming AE proposed capsules instead of neurons to keep the maximum information, e.g. pose and velocity. However, the idea did not work efficiently until the introduction of the Routing by Agreement algorithm in 2017(Sabour et al., 2017), which outperforms CNNs in some databases and shows impressive results.

This paper highlights the limitations of CNNs and the high performance of CapsNets in diverse implementations. We present a variety of selections of the best performing works in CapsNets from various viewpoints. We compare different CapsNets' models, and we discuss their benefits and challenges. This survey is done after consulting other similar papers. We believe that our review presents the most recent works in this field. It gives a clear view of CapsNets' series and updates, and it explores a possible future scope of research.

This paper is organized as follows: In Section 2, we introduce CNNs and their limitations. Then, we detail Capsule Network architecture and its progress in Section 3. Furthermore, we present implementations' domains and fields of this Deep Learning (DL) network in Section 4. After that, we describe CapsNets updates in Section 5. The series and shortcomings of Capsule Networks are described in Section 6. Finally, we conclude in Section 7.

2 CONVOLUTIONAL NEURAL NETWORKS

Convolutional Neural Networks (CNNs) are very powerful in image classification and processing(Q. Zhang et al., 2016)(Krizhevsky et al., 2012).They are considered state-of-the-artin computer vision and widely used in object recognition systems(Maturana & Scherer, 2015) and self-driving cars(Jung et al., 2016).

2.1 Overview of CNNs

CNNs treat an input image by four kinds of layers: convolutional layers, pooling layers, flattening layers and fully connected layers. Convolutional layers apply multiple kernels to the input and activate the output according to the rectified linear activation function (ReLU)(He et al., 2015)to generate a features map(equation 1).The pooling generates a pooled feature map using Max-pooling (equation 2), which chooses the most important pixels to be passed to the next layer.Therefore, it reduces the dimension of images. These two layers are repeated several times to refine feature extraction. Next, the flattening layerflattens the pooled feature map into a column matrix. This matrix will be passed toa Fully Connected (FC) artificial neural network that consists of an input layer, hidden layers and output layer. Figure 1 shows the CNNs' structure.

$$X'_{1,1,1}=ReLU(X_{1,1}*k_{1,1}+X_{1,2}*k_{1,2}+X_{2,1}*k_{2,1}+X_{2,2}*k_{2,2}) \quad (1)$$

$$P_{1,1}=max(X'_{1,1,1};X'_{1,1,2};X'_{1,2,1},X'_{1,2,2}) \quad (2)$$

The convolution moves by a number of steps called strides, from left to right and from top to bottom on the input to generate the feature map. To preserve a maximum of features, several distinct kernels are applied to the input to obtain corresponding feature maps. The ReLU function is for adding nonlinearity into the model. Max-pooling scans each feature map, and selects the maximum value according to filter size, and creates a pooled feature map.

2.2 CNNs Shortcomings

Convolutional Neural Networks were introduced two decades ago. Through all these years, CNNs were widely developed and adjusted. However, they still have some shortcomings:

- Inability to understand data structure(Hosseini et al., 2017): CNNsare not interested in position properties and hierarchical structures i.e. relations between objects' parts. Max-pooling reduces the dimension of images and causes a loss of some useful features.
- Inabilityto be spatially invariant:CNNs are only invariant to translation, but if the input images have been reversed, rotated or tilted the performance decreases drastically. They are unable to detect deformation, pose and texture of an image (Sabour et al., 2017).
- Viewpoint variance: different viewpoints of an object lead to changes in neural activities. Hence, to recognize objects,the network should learn different variations of the images.That requires a lot of training data and a long training time.
- Overfitting: when the cameraor the illumination of the image is changed, CNNs cannot perform well (Ahmadvand et al., 2016).
- Sensitive to adversarial attacks(Su et al., 2019): CNNs can easily be fooled by adding some carefully constructed noise to the input image.

3 CAPSULESNETWORK PROGRESS

The idea of Capsule Network was introduced in 2011 to overcome the shortcomings of CNNs regarding robustness. It has been tested on highly complex data and showed a high performance. The following sub-chapters describe the main milestones in the progress of CapsNets.

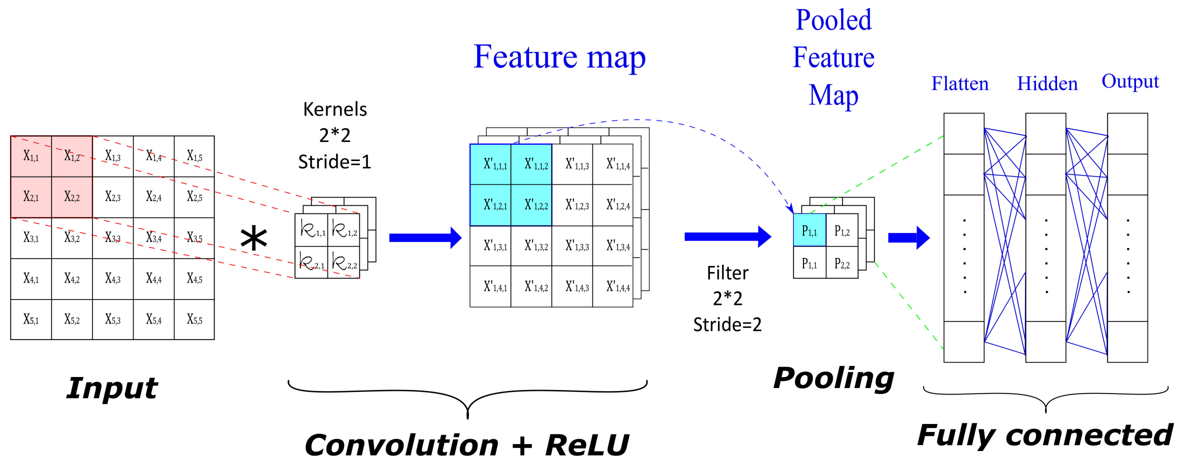

Figure 1: CNNs structure with one Convolution+ReLU layer.

3.1 Transforming Auto-encoders

Transforming Auto-encoders (TAEs)(Hinton et al., 2011) were the first seed of capsule networks. TAEs are Auto-encoders that apply a transformationmatrix to the extracted features' pose, so the network can be trained to predicttransformations like rotation, scaling and translation.

Unlike CNNs thatare only invariant to translation, TAEs are equivariant. This property makes them understand proportion change and adjust themselves accordingly to keep the features' pose information. Equivariance is achieved in these Auto-encoders by using vectors to represent objects, where each vector contains scalar values that represent the instantiation parameters of the object.

TAEsconsist of several capsules, where each capsule is a group of neurons that represent an object or a part of an object in a specific location using inverse rendering.They extract instantiation parameters from the image to draw it again.

ATAE's capsule is composed of recognition units and generative units. The output of each capsule represents the contribution to reconstruct the output image. Figure 2 details the structure of the TAEs.

Recognition units (blue circles in Figure 2) detect pose parameters represented by matrix A and computeP, the probability that the capsule's feature is present in the image. Then, the capsule will transfer these values to the generative units layer.

Generative units (red circles in Figure 2) are fed with **TA**, where **T** is the transformation matrix.These units compute the capsule's contribution to the transformed image and multiply it by the probability **P**. Finally, all capsules' contributions are combined to reconstruct the output image.However, this architecture could not work properly in 2011, because of computer hardware limitations and the absence of efficient algorithms.

3.2 Dynamic routing between capsules

In 2017,(Sabour et al., 2017) succeeded to implement an efficient algorithm to relate capsules, that showed better performance than CNN on the MNIST dataset. It is called Dynamic Routing Between Capsules or Routing by Agreement between capsules (RBA). This paper (Sabour et al., 2017) was the official definition of CapsNets as a network of capsules.The output of a capsule is called activation or instantiation vector. The length of this vector represents the probability that the feature actually exists. The orientation of the vector encodes the feature's instantiation parameters, i.e. thickness,localization, width and so on. The CapsNets Encoder consists of three main parts:

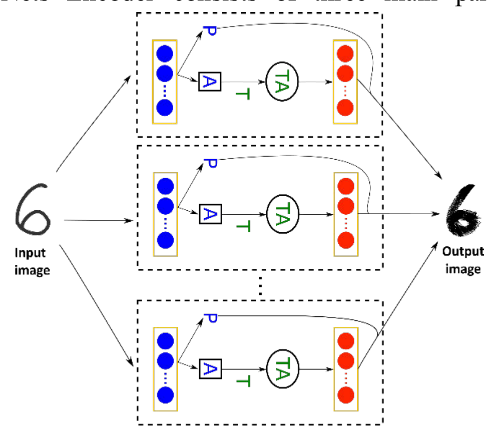

Figure 2: Transforming Auto-encoders' capsule structure.

Convolutional layer, PrimaryCaps layer and ClassCapslayer (also called as DigitCaps) (Figure 3). The Convolutional layer extracts image features through convolution kernelsto result in a feature map. Then, a ReLU function is applied to provide non-linearity and to activate the feature map values. The output feature map is scanned another time bykernels and generates a new feature map. PrimaryCaps group the generated features to vectors to create primary capsules. Finally, the PrimaryCaps are routed to the ClassCaps layer by Dynamic Routing Between Capsules (Algorithm 1). The contribution of each capsule***i*** in PrimaryCaps to each capsule ***j*** in ClassCaps is computed as follows:

$$\hat{u}_{ji} = W_{ij} u_i \quad (3)$$

Where u_i is the output of capsule i, and $\hat{u}_{ji}$is a prediction vector. W_{ij} is a weight matrix.

Each capsule j in ClassCaps computes the total prediction vector $\mathbf{s}_j$(equation 4). To ensure that the vector length is between 0 and 1, a squashing function is applied (equation 5), which does not affect the instantiation parameters.

$$\mathbf{s}_j = \sum_{i=1}^{n} c_{ij} \hat{u}_{ji} \quad (4)$$

$$V_j = \frac{||s_i||^2 \, s_i}{1+||s_i||^2 \, ||s_i||} \quad (5)$$

c_{ij} is the coupling coefficient determined by a SoftMax function (equation 6). This coefficient is used by Dynamic Routing to determine the relation between low-level and high-level capsules through repetitive routing. The agreement between capsules is reflected by the product of the prediction vector and a coupling coefficient. If the agreement is high, the low-level capsule and the high-level capsule are related to each other and the coupling coefficient will increase otherwise, it will decrease. Notice that $c_{ij}$is updated in this step by updating b_{ij}(equation 7), unlike W_{ij} that are updated by backpropagation.

$$c_{ij} = \frac{e^{b_{ij}}}{\sum_k e^{b_{ik}}} \quad (6)$$

$$b_{ij} = b_{ij} + V\hat{U} \quad (7)$$

The Decoder part (Figure4) aims to reconstruct the input image, it is made up of three Fully connected (FC) layers that generate output which is reshaped toa grayscale image.

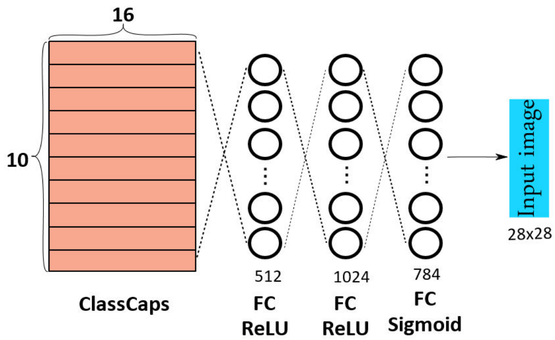

Figure 3: CapsNetsDecoder.

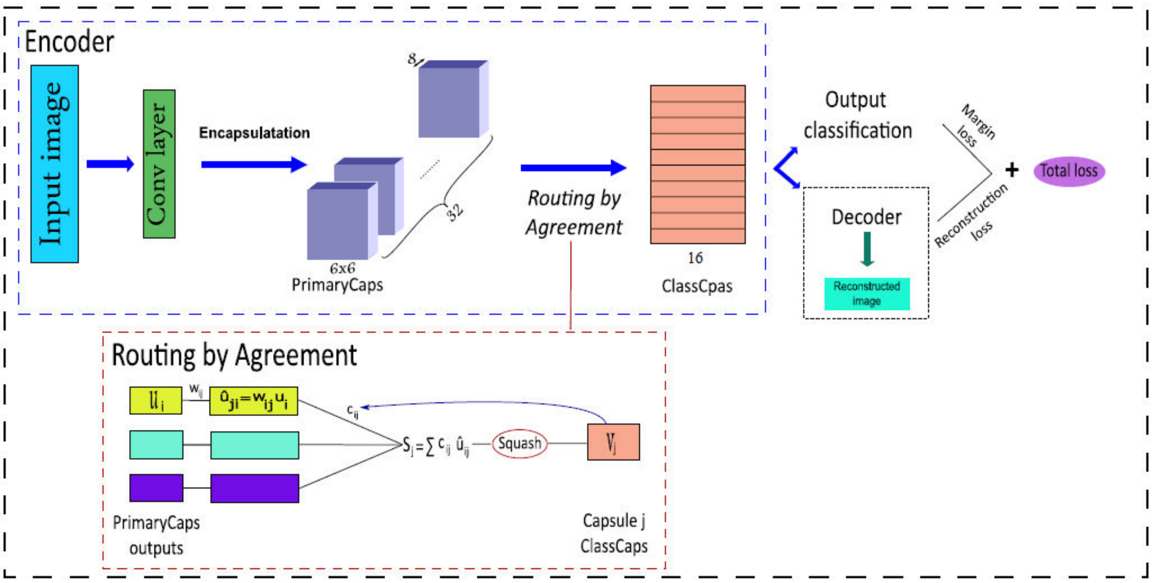

Figure 4: CapsNets Encoder, Decoder, Routing by Agreement.

As long as the CapsNets consist of classification and reconstruction part, the total lossTL will be calculated on two halves: (i) The first one punishes incorrect classifications L_k (encoder-part), (ii) and the second punishes reconstruction errorD (decoder-part) by mean square loss. The following equation represents the margin loss of classification:

$$L_k = E_k \max(0, t^+ - \|v_k\|)^2 + \lambda(1 - E_k)\max(0, \|v_k\| - t^-)^2 \qquad (8)$$

Where $E_k \max(0, t^+ - \|v_k\|)^2$ is calculated if an object of class k is present withE_k is set to 1 , and $\lambda(1 - E_k)\max(0, \|v_k\| - t^-)^2$ is calculated for the opposite case with $E_k = 0 . t^+ = 0.9$ and $t^- = 0.1$ are set to prevent the length to max out or collapse the loss function unreasonably, λ is set to 0.5 to control the down weighting of initial weights from influencingmodeldecisions. This entity loss (L_k) is then summed withthe reconstruction loss (equation 9) tocompute the total loss (equation 10), which is used to evaluate the performance of the capsule model.

$$D = \text{MSELoss}(y, y') \qquad (9)$$

y is the input image and y′ is the reconstructed image

$$TL = L_k + \alpha D \qquad (10)$$

α is the down-scaling factor(taken as 0.0005) used to prevent the D loss from dominating over the L_k loss.

3.3 Matrix Capsules with EM Routing

In (Hinton et al., 2018),another algorithm was proposed for routing between capsules called Expectation Maximization Routing (EMR). Unlike RBA's capsules that use elements' vectors to represent the pose of an object and the vectors' lengths to represent the probability of existence, EMR capsules use pose matrix and activation probability separately.Expectation Maximization is a clustering algorithm that clusters datapoints into Gaussian distribution, with each cluster defined by (μ:mean,σ: standard deviation).In capsule network, EMR groups capsules into a parent capsule.The high-level capsule is activated if there is an agreement among votes from low-level capsules. The low-level capsule makes votes (predictions) on the pose matrices of its potentialparent capsule. The

	RBA	EM for RBA
Algorithm	i: capsule in layer l j: capsule in layer l+1 Algorithm 1 Dynamic Routing(Sabour et al., 2017) procedure ROUTING($\hat{u}_{ji}$, r, l) $\forall b_{ij}$, $b_{ij} \leftarrow 0$ For k iterations do $c_{ij} \leftarrow$ SoftMax(b_{ij}) equation 4 $s_j = \sum c_{ij}\hat{u}_{j\|i}$ Vi=squash(s_j) equation 3 $b_{ij} = b_{ij} + V\hat{U}$ Return V_j	Ω_L capsules of the layer l Algorithm 2 EM Routing(Hinton et al., 2018) Procedure EM ROUTING(a,V) $\forall i \in \Omega_L$, $j \in \Omega_{L+1}$: $R_{ij} \leftarrow 1/\|\Omega_{L+1}\|$ For t iterations do $\forall j \in \Omega_{L+1}$: M-Step(a,R,V,j) $\forall i \in \Omega_L$: E-Step(μ,σ,a,V,i) Return a,M M-Step: updates (μ,σ,a) based on R the assignment probability E-Step: recalculates R based on new μ,σ and a
Properties	- The representation of a capsule's input and output is a vector. - The probability of existence is represented by the length of a vector. - Squashing function for probability. - Prediction vector: $\hat{U}_{ji} = W_{ij} U_i$. - Returns: Probability (V). - Coupling coefficient: C. - Margin loss + MSELoss.	- New parameter: capsule's pose matrix: M. - The representation of a capsule's input and output is a matrix. - The probability of the presence of an entity is represented by a parametera(activation probability). - Gaussian probability. - Vote:$V_{ij} = M_i W_{ij}$ - Returns: Activation probability (a)+ Pose matrix (M). - Assignment probability: R quantifies the runtime connection between child capsule and its parent capsule. - Spread loss: maximizes directly the divide between the wrong class's activation and target one.

Table 1: Difference between RBA and EMR.

vote (V) is calculated by multiplying the pose matrix(M) ofthe low-level capsulewith aviewpoint invariant transformation(W).

$$V = MW \quad (11)$$

In EMR, the representation of a capsule's input and output are matrices instead of vectors. Moreover, the likeliness of the existence of an entity is presented by the activation probability ***a***instead of a length vector. The probability is computed without usinga squashing function, which is considered "not objective and sensible"(Hinton et al., 2018).Table 1 clarifies the difference between Dynamic Routing algorithm 1 and EMR algorithm 2.

3.4 Stacked Capsule Auto-encoders

In 2019,(Kosiorek et al., 2019) introduced an unsupervised capsule Auto-encoder called Stacked Capsule Auto-encoders (SCAEs). This capsule network uses objects to predict parts, in contrast to EM Routing and Routing by Agreement that use a part-whole relationship to predict the presence of the object. The inference routing used in both previous works is inefficient and it is discussed in further research (Li et al., 2018; S. Zhang et al., 2018),while SCAEs amortized this inference.

The SCAEs consist of two stages. In the first stage called Part Capsule Auto-encoder (PCAE), the model predicts presences and poses of part templates directly from the image and tries to reconstruct the image by appropriately arranging the templates. In the second stage called Object Capsule Auto-encoder (OCAE), the model organizes discovered parts and their poses into a smaller set of objects. These objects reconstruct the part poses using a separate mixture of predictions for each part.

SCAEsare the only method that achieves competitive results in unsupervised object classification without relying on mutual information (MI).

4 IMPLEMENTATIONS

CapsNets showed their performance in various fields such as medical or chemical image recognition, audio recognition, sentiment analysis and many others.

These kinds of networks have the best performance in detecting spoof attacks. (Nguyen et al., 2019) applied capsule network to the forensics task. It is used to detect various kinds ofspoofs from replay attacksusingprinted images or recorded videos to computer-generated videos. Furthermore, the RBA algorithm used improves detection performance on complex and almostperfectly forged images and videos. Itshowed agreat performance and had perfect accuracy at frame level and video level dataset.

Capsule networks havealso proven their efficiency in the 3D domain. In (Yongheng Zhao et al., 2019), they are used to treat sparse 3D point clouds.They preserve spatial arrangements of the input data with good learning ability and generalization properties. The model performs well under rotation, part-segmentation and 3D reconstructionand it has a low reconstruction error.

(Duarte et al., 2018) introduce a 3D capsule network for action detection in videos, by introducing capsule-pooling with skip connections in the convolutional layer to decrease capsule routingcomputation.

In the medical domain(Mobiny & Nguyen, 2018),capsules have also been developed to handle characteristics of 3D lung nodule classification, and speed up CapsNets by factor three by a consistent RBA mechanism. The proposed dynamic routing mechanism consists of enforcing all capsules in the Primary Capsule layer referring to the same pixels to have the same coupling coefficient, which reduces the number of routing coefficients and speeds up the model while keeping the accuracy of the original CapsNets.

1D-CapsNet (Butun et al., 2020)has been introducedfor automated detection of coronary artery disease (CAD) from ECG signals (electrocardiography-signal).Even though the model achieved a high accuracy it needs to overcome the long of training time. Furthermore, the model needs a large dataset for training. This issue could be addressed by few-shot learning (Ren et al., 2020).

CapsCarcino is another implementation of capsules in medicine (Y.-W. Wang et al., 2020). It has been introducedto distinguish between carcinogens and noncarcinogens.This capsule network is very helpful for carcinogen risk assessment in drugs.CapsCarcinois very robust for small-sized sparse datasets: with just 20% of the dataset, it performs comparably to the other methods using the full training dataset.

WB-Caps (Baydilli & Atila, 2020) is a capsule network architecture that classifieswhite blood cells into five categories. WB-Caps can help to interpret the patient's condition by performing blood tests with little cost, based on some characteristics of WBCs like ratio or shape.The model obtained a high accuracy without over-fitting.

CapsNet-static-routing (Kim et al., 2020) is a CapsNets model used for text classification.It shows a high performance and stable results even after adding random noise to the dataset, the result does not change, and sentences keeptheir meaning. The experimental results of the classification indicate that the accuracy of the staticrouting is higher than the dynamic one. Moreover, the model has a shorter trainingtime than the original CapsNets. On the other hand, due to the high variability in text, CapsNet-static-routing is not robust enough for document classification as opposed to image classification. It needs to be flexible for text modifications, like word order shuffling.

(Lei et al., 2020) introduced Attention-Based Capsule Network (ACN) for Tag Recommendation. The model is based on the capsule network with Dynamic Routing plus an attention mechanism.The model is flexible to be applied for image and video tagging, too. Moreover, ANC could be improved by using Expectation Maximization routing, where pose matrix might extract more information and give better tag results.

Forintelligent faultdiagnosis, Capsule Auto-encoder (Ren et al., 2020) (CaAE) has been proposed to resolve theproblems of traditional and modern intelligent fault diagnosis:the need of a large set of samples of faults and the need of diagnosis models to possess the ability of quick updating. The ability of CaAE to extract and fuse featuresreduces the dependence on the number of samples and training time, whichmakesCaAE suitable for few-shot learning without overfitting. The modelis very robust under noisy datasets and it shows higher accuracy, less training time and a smaller number of epochs compared to methods in(J. Wang et al., 2019) and (Jia et al., 2016).

5 CAPSNETSUPDATES& IMPROVEMENTS

(Nguyen et al., 2019)proposed CAPSULE-FORENSICS toimprove the algorithm of (Sabour et al., 2017).A Gaussian random noise has been added to the weight tensor to reduce over-fitting, andan additional squash has been applied before routing by iterating to keep the network more stable.

(Kim et al., 2020) suggest a static routing method instead of dynamic routing and ELU-gate(Dauphin et al., 2017) instead of pooling. Static routing reduces the computational complexity of dynamic routing. ELU-gate method selects which neurons to activate without losing spatial information.

(Rajasegaran et al., 2019)havegone deep with capsule network (Deepcaps) using the concept of skip connections and 3D convolutions to build a 3D convolution system based on the dynamic routing algorithm. Skip connections within a capsule cell allow good gradient flow in backpropagation, and 3D convolution reduces the number of parameters.The original CapsNetsdecoder(Sabour et al., 2017) has been replacedbya Deconvolutional decoder, which strengthens the use of reconstruction loss as a regularization term. This decoder is better at reconstructing spatial relationships and at regularizing capsules.

(Phong & Ribeiro, 2019) introduced two advanced models (Capsule 32 V1 for images 32*32 pixels and Capsule 32 V2 for images of 64*64pixels) to improve CapsNets by expanding more pooling layers to filter image backgrounds and more reconstruction layers to allow better image restoration.Both modelsshowed a good performance but theyare more sensitive to changes.

To reduce epistemic and the homoscedastic uncertainty, (Ramírez et al., 2020)present a Bayesian formulation of Capsule networks (BCN). They hybridized Deep Bayesian Neural Networks (DBNN)(Zhu & Zabaras, 2018)with Capsule Networks.The model attainedgood results with less uncertainty and less error due to performing dropout and including the homoscedastic uncertainty in the loss function and usinga regularization term over the linear transformations in the inverse graphics.

As it has been introduced in the RBA algorithm, the SoftMax activation function is used to compute the coupling coefficient $\mathbf{c}_{ij}$.(Z. Zhao et al., 2019) demonstrated that SoftMax prevents CapsNets to find the optimal coupling to route between low-level and high-level capsules. After multiple routing iterations, it often leads to uniform probabilities. For that, SoftMax has been replaced by the Max-Min normalization.This normalization reduces the test error to 0.17% on MNIST and allows to increase the number of routing iterations without overfitting.

To reduce CapsNets parameters(Yi et al., 2019) designed the CapsNetPr network that uses a pooling method, decomposition and sharing of the transformation matrix to address this issue. As aresult,the CapsNets parameters have been reduced significantlyacross different datasets while keeping the performance of CapsNets.

6 CAPSULE NETWORKS SERIES, ADVANTAGES AND SHORTCOMINGS

Capsule Networks are used for treating various kinds of data such as images, text, videos. The variety of data requires some modifications on the original network structure. Table 2 summarizes the CapsNets series.

The majority of CapsNets research papers worked on the RBA algorithm, eitherin the original.

Paper	Proposed model	Task	Characteristics	Additions	Dataset	Accuracy on proposed model (%)/ Metric	Baseline model	Accuracy on baseline model (%)/Metric
Capsule-forensics: Using Capsule Networks to Detect Forged Images and Videos (Nguyen et al., 2019)	**CAPSULE-FORENSICS**	Spoofs detection	Has the best performance and accuracy at frame level and video level dataset.	VGG-19 layer before the primary layer. Addition of Gaussian noise to the weight matrix. Application one additional squash before RBA.	**Deepfake dataset**	99.23%	MesoInception-4 Meso-4	98,4% 96.90%
DeepCaps: Going Deeper with Capsule Networks (Rajasegaran et al., 2019)	**DeepCaps**	Image classification	Surpasses the CapsNets' results on CIFAR10, SVHN and Fashion MNIST. Reduces the number of parameters.	Skip connections within capsule cells. 3D convolution CapsCells. Class-independent decoder.	**CIFAR10** **SVHN** **F-MNIST** **MNIST**	CIFAR10: 92.74% SVHM: 97.56% F-MNIST: 94.73% MNIST: 99.75%	RBA	CIFAR10: 89.40% SVHM: 95.70% F-MNIST: 93.60% MNIST: 99.75%
3D Point Capsule Networks (Yongheng Zhao et al., 2019)	**3D-PointCapsNet**	3D points clouds process	A higher accuracy compared with AltasNetand and smaller training-set.	3D Capsule-Encoder. 3D Capsule-Decoder.	**ShapeNet 55**	89.3%	Latent-GAN FoldingNet	85.7% 88.4%
1D-CADCapsNet: One dimensional deep capsule networks for coronary artery disease detection using ECG signals (Butun et al., 2020)	**1D-CADCapsNet**	Detection of CAD ECG signals	High performance using raw ECG signals without any feature extraction/selection or QRS detection.	Redefinition of layers' parameters. Addition of some sub-layers to detect CAD ECG signal segments: tow 1D-Conv before primary caps then ECG caps.	**ECG dataset**	2 second ECG segments: 99.4% 5 second ECG segments: 98.6%	CNN CNN-LSTM	2 second ECG segments 94.95% 5s second ECG segments 95.11% 5s second ECG segments: 99.85%
Text Classification using Capsules (Kim et al., 2020)	**CapsNet-static-routing**	Text Classification	Higher performance and noise-robustness compared to the state-of-the-art methods of text classification.	Static routing. ELU-gate instead of pooling. Removal of the coupling coefficient used in RBA.	**Sentences from TREC-QA test data**	74%	Dynamic Routing	65%

A model with the ability of few-shot learning and quick updating for intelligent fault diagnosis (Ren et al., 2020)	**CaAE**	Intelligent fault diagnosis	The ability of few-shot learning. Rapid updating and the ability to resist noise.	Combination of AE and CapsNets, which is composed of three parts: feature extraction, feature fusion and fault diagnosis.	**motor bearings provided data**	99.85%	SEFAM BNSAEs BNAE	99.07% 97.65% 98.53%
CapsCarcino: A novel sparse data deep learning tool for predicting carcinogens (Y.-W. Wang et al., 2020)	**CapsCarcino**	Molecules classification	Higher accuracy compared with SVM, RF, KNN, XGBoost, CNN. Robust for small size sparse dataset.	Two convolutional layers, one fully connected layer, one PrimaryCaps layer and one ToxCaps layer.	**Carcinogenic Potency Database (CPDB)**	81.8%	SVM RF kNN XGBoost CNN	70.0% 64.2% 65.7% 59.6% 66.8%
Classification of white blood cells using capsule networks (Baydilli & Atila, 2020)	**WBCaps**	White blood cells classification	High performance compared with Deep Learning methods and medical analysis techniques.	Optimization of hyper-parameters usingthe "babysitting" method. PReLU function for convolutional and for FC layer.	**LISC dataset**	96.86%	Inception-ResNETv2 Inceptionv3 ResNET50 VGG19	82.50% 80.00% 80.00% 77.50%
Bayesian capsule networks for 3D human pose estimation from single 2D images (Ramírez et al., 2020)	**Bayesian CapsNet**	3D pose estimation from a single 2D image	Reduces the homoscedastic uncertainty.	Bayesian Capsules. Bayesian FC neurons. Dropout of initial capsules. Regularization term over the linear transformations in the inverse graphics.	**Human3.6M dataset**	Error (mm.):71.7	Tome (Tome et al., 2017) Rogez(Rogez et al., 2019)	Error (mm.) 79.6 56.5
Tag Recommendation by Text Classification with Attention-Based Capsule Network (Lei et al., 2020)	**Attention-based CapsNets (ACN)**	Tag Recommendation	Outperforms the standard capsule networks. Flexibility to be applied for image and video tagging	Architecture: Embedding layer, attention layer, convolutional layer, primary capsule layer, Fully connected layer, dropout layer.	**TPA from AMiner** **AG from ComeToMyHead**	TPA: macro-P 0.829 macro-R 0.825 macro-F1 0.824 AG: macro-P 0.926 macro-R 0.922 macro-F1 0.923	CapsNets	TPA: macro-P 0.820 macro-R 0.815 macro-F1 0.814 AG: macro-P 0.921 macro-R 0.918 macro-F1 0.920

Table 2: CapsNet series.

implementation or in improvement, while EMR and SCAE did not get the same attention from researchers. Just like with the use of CapsNets in the 3D domain, only a few works have been focused on this field(Yongheng Zhao et al., 2019),(Duarte et al., 2018), (Weiler et al., 2018), (Jiménez-Sánchez et al., 2018; Mobiny & Nguyen, 2018).

6.1 Advantages

CapsNetsarea very promising Deep Learning model, whichhas the capacity of learning relationships among image objects. This architecture has so many positive aspects:

- Viewpoint invariance (Hinton et al., 2011).
- The dynamic routing algorithm extracts more meaningful features compared to CNNs (Sabour et al., 2017).
- They are equivariant, they are unaffected by positional changes.
- They efficiently classify small data sets without data augmentation (Su et al., 2019),(Y.-W. Wang et al., 2020).
- They are more robust than traditional CNNs to white box adversarial attacks (Hinton et al., 2018)
- EMR achieved higher accuracy than the state-of-the-art CNN on the smallNORB dataset (Hinton et al., 2018).
- They are robust to an imbalanced class distribution (Jiménez-Sánchez et al., 2018).
- They increase the certainty to recognize the pose of an object since RBA and EMR activate a capsule after comparing several incoming pose vectors.

These characteristics make CapsNets more powerful compared to other DL approaches in terms of generalization capability, accuracy, required training time and robustness to viewpoint changes.

6.2 Shortcomings

From RBA to Stacked Capsule Auto-encoder, CapsNetshave showngood performance in different domains likein image classification,signal treatment, pose extraction, text classification and many other tasks. They are applicable to various kinds of datasets by adapting the architecture or the learning algorithmto the specificity of the data. Nevertheless, Capsule Networks suffer some drawbacks. Routing by agreement is not optimal for document classification, unlike for image classification, due to the high variability in a text(Kim et al., 2020).

Although the CapsNets showed an impressive result in the MNIST dataset and did well on SVHM, they still perform poorly on CIFAR10, even when going deep in the Capsule network by DeepCaps(Rajasegaran et al., 2019), achieving an error of 8,99%, which is higher than the error rate of the current state-of-the-art 3,47%. The higher error rate can be explained with the complexity of the background and the intra-class variation of CIFAR10.

Adownsideofthe treated network is the high number of parameters to be trained(School of Computing, Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P.R. China et al., 2019). With a small inputimageof28x28, the original CapsNets architecture needs 8,2 M training parameters. More than half of these parameters come from the PrimaryCaps layer that executes reshaping and dynamic routing operations. The larger the images to be processed become, the greater becomes the number of parameters to be trained. Deepcaps(Rajasegaran et al., 2019)managed to reduce the number of parameters by 68%, while (Xiong et al., 2019; Yi et al., 2019) used a pooling method which loses meaningful information.

The learning process in CapsNets is slow due to the routing process that requires a loop to refine the coupling coefficient. Moreover, CapsNets require more computational resources since the outputs of primary capsules are activity vectors rather than scalars, which requires more memory.

7 CONCLUSION AND DIRECTIONS FOR FUTURE WORK

In this paper, Capsule networks have been introducedwith their main progress steps: Transforming Auto-encoders, Routing by Agreement Between Capsules,Matrix capsules with EM routing and Stacked Capsule Auto-encoders. The advantages of grouping extracted features into capsules to keep all input information have been explained as well as learning algorithms, architecture and CapsNets series. Capsule networks guarantee equivariant properties which make the network robust when undergoing a transformation. Furthermore, CapsNets achieved a very promising result with a small training dataset and without overfitting. However, they need to be improved to perform well with multi-class data and complex data such as CIFAR10. This Deep Learning Networks need more experiments, searches and tests to explore their maximum capacity. Besides, more attention for the EM Routing and SCAE are necessary to make them more powerful and applicable in different datasets and to realizethe full potential of CapsNets.

New insights could be provided from going deep with Matrix capsules with EM routing and Stacked Capsule Auto-encoders as advanced CapsNets, also from working on reducing the complexity of these models and combining Capsule networks with other Deep Learning methods. Furthermore, self-driving cars can take advantage of the CapsNets' accuracy and robustness against transformations made on inputs to trick the network. Moreover, the unsupervised learning used in Stacked Capsule Auto-encoders will be useful to solve complex reinforcement learning tasks.

REFERENCES

Ahmadvand, P., Ebrahimpour, R., & Ahmadvand, P. (2016). How popular CNNs perform in real applications of face recognition. *2016 24th Telecommunications Forum (TEL FOR)*, 1–4.

Baydilli, Y. Y., & Atila, Ü. (2020). Classification of white blood cells using capsule networks. *Computerized Medical Imaging and Graphics*, *80*, 101699.

Butun, E., Yildirim, O., Talo, M., Tan, R.-S., & Rajendra Acharya, U. (2020). 1D-CADCapsNet: One dimensional deep capsule networks for coronary artery disease detection using ECG signals. *Physica Medica*, *70*, 39–48.

Dauphin, Y. N., Fan, A., Auli, M., & Grangier, D. (2017). Language modeling with gated convolutional networks. *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, 933–941.

Duarte, K., Rawat, Y., & Shah, M. (2018). VideoCapsuleNet: A Simplified Network for Action Detection. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 31* (pp. 7610–7619).

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. *2015 IEEE International Conference on Computer Vision (ICCV)*, 1026–1034.

Hinton, G. E., Krizhevsky, A., & Wang, S. D. (2011). Transforming Auto-Encoders. In T. Honkela, W. Duch, M. Girolami, & S. Kaski (Eds.), *Artificial Neural Networks and Machine Learning – ICANN 2011* (Vol. 6791, pp. 44–51). Springer Berlin Heidelberg.

Hinton, G. E., Sabour, S., & Frosst, N. (2018). Matrix capsules with EM routing. *6th International Conference on Learning Representations, ICLR*, 1–15.

Hosseini, H., Xiao, B., Jaiswal, M., & Poovendran, R. (2017). On the Limitation of Convolutional Neural Networks in Recognizing Negative Images. *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 352–358.

Jia, F., Lei, Y., Lin, J., Zhou, X., & Lu, N. (2016). Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. *Mechanical Systems and Signal Processing*, *72–73*, 303–315.

Jiménez-Sánchez, A., Albarqouni, S., & Mateus, D. (2018). Capsule Networks Against Medical Imaging Data Challenges. *Intravascular Imaging and Computer Assisted Stenting and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis*, 150–160.

Jung, S., Lee, U., Jung, J., & Shim, D. H. (2016). Real-time Traffic Sign Recognition system with deep convolutional neural network. *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, 31–34.

Kim, J., Jang, S., Park, E., & Choi, S. (2020). Text classification using capsules. *Neurocomputing*, *376*, 214–221.

Kosiorek, A., Sabour, S., Teh, Y. W., & Hinton, G. E. (2019). Stacked Capsule Autoencoders. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d\textquotesingle Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 32* (pp. 15512–15522).

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems 25* (pp. 1097–1105).

Lee, H., Ekanadham, C., & Ng, A. Y. (2008). Sparse deep belief net model for visual area V2. In J. C. Platt, D. Koller, Y. Singer, & S. T. Roweis (Eds.), *Advances in Neural Information Processing Systems 20* (pp. 873–880).

Lei, K., Fu, Q., Yang, M., & Liang, Y. (2020). Tag Recommendation by Text Classification with Attention-Based Capsule Network. *Neurocomputing*.

Li, H., Guo, X., DaiWanli Ouyang, B., & Wang, X. (2018). Neural Network Encapsulation. *Proceedings of the European Conference on Computer Vision (ECCV)*, 252–267.

Maturana, D., & Scherer, S. (2015). VoxNet: A 3D Convolutional Neural Network for real-time object recognition. *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 922–928.

Mobiny, A., & Nguyen, H. V. (2018). Fast CapsNet for Lung Cancer Screening. *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, 741–749.

Nguyen, H. H., Yamagishi, J., & Echizen, I. (2019). Capsule-forensics: Using Capsule Networks to Detect Forged Images and Videos. *ICASSP 2019*

- *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2307–2311.

Phong, N. H., & Ribeiro, B. (2019). Advanced Capsule Networks via Context Awareness. *International Conference on Artificial Neural Networks. Springer, Cham*, *11727*, 166–177.

Pu, Y., Gan, Z., Henao, R., Yuan, X., Li, C., Stevens, A., & Carin, L. (2016). Variational Autoencoder for Deep Learning of Images, Labels and Captions. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 29* (pp. 2352–2360).

Rajasegaran, J., Jayasundara, V., Jayasekara, S., Jayasekara, H., Seneviratne, S., & Rodrigo, R. (2019). DeepCaps: Going Deeper With Capsule Networks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10717–10725.

Ramírez, I., Cuesta-Infante, A., Schiavi, E., & Pantrigo, J. J. (2020). Bayesian capsule networks for 3D human pose estimation from single 2D images. *Neurocomputing*, *379*, 64–73.

Ren, Z., Zhu, Y., Yan, K., Chen, K., Kang, W., Yue, Y., & Gao, D. (2020). A novel model with the ability of few-shot learning and quick updating for intelligent fault diagnosis. *Mechanical Systems and Signal Processing*, *138*, 106608.

Rogez, G., Weinzaepfel, P., & Schmid, C. (2019). LCR-Net++: Multi-person 2D and 3D Pose Detection in Natural Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1.

Sabour, S., Frosst, N., & Hinton, G. E. (2017). Dynamic Routing Between Capsules. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 30* (pp. 3856–3866). Curran Associates, Inc.

School of Computing, Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P.R. China, Jiang, X., Wang, Y., Liu, W., Li, S., & Liu, J. (2019). CapsNet, CNN, FCN: Comparative Performance Evaluation for Image Classification. *International Journal of Machine Learning and Computing*, *9*(6), 840–848.

Su, J., Vargas, D. V., & Sakurai, K. (2019). Attacking convolutional neural network using differential evolution. *IPSJ Transactions on Computer Vision and Applications*, *11*(1), 1.

Tome, D., Russell, C., & Agapito, L. (2017). Lifting from the Deep: Convolutional 3D Pose Estimation from a Single Image. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5689–5698.

Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. *Proceedings of the 25th International Conference on Machine Learning - ICML '08*, 1096–1103.

Wang, J., Li, S., An, Z., Jiang, X., Qian, W., & Ji, S. (2019). Batch-normalized deep neural networks for achieving fast intelligent fault diagnosis of machines. *Neurocomputing*, *329*, 53–65.

Wang, Y.-W., Huang, L., Jiang, S.-W., Li, K., Zou, J., & Yang, S.-Y. (2020). CapsCarcino: A novel sparse data deep learning tool for predicting carcinogens. *Food and Chemical Toxicology*, *135*, 110921.

Weiler, M., Geiger, M., Welling, M., Boomsma, W., & Cohen, T. S. (2018). 3D Steerable CNNs: Learning Rotationally Equivariant Features in Volumetric Data. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 31* (pp. 10381–10392). Curran Associates, Inc.

Xiong, Y., Su, G., Ye, S., Sun, Y., & Sun, Y. (2019). Deeper Capsule Network For Complex Data. *2019 International Joint Conference on Neural Networks (IJCNN)*, 1–8.

Yi, S., Ma, H., & Li, X. (2019). Modified Capsule Network for Object Classification. In Yao Zhao, N. Barnes, B. Chen, R. Westermann, X. Kong, & C. Lin (Eds.), *Image and Graphics* (pp. 256–266). Springer International Publishing.

Zhang, Q., Yang, L. T., & Chen, Z. (2016). Deep Computation Model for Unsupervised Feature Learning on Big Data. *IEEE Transactions on Services Computing*, *9*(1), 161–171.

Zhang, S., Zhou, Q., & Wu, X. (2018). Fast Dynamic Routing Based on Weighted Kernel Density Estimation. *Cognitive Internet of Things: Frameworks, Tools and Applications*, 301–309.

Zhao, Yongheng, Birdal, T., Deng, H., & Tombari, F. (2019). 3D Point Capsule Networks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1009–1018.

Zhao, Z., Kleinhans, A., Sandhu, G., Patel, I., & Unnikrishnan, K. P. (2019). Capsule Networks with Max-Min Normalization. *ArXiv:1903.09662 [Cs]*. http://arxiv.org/abs/1903.09662

Zhu, Y., & Zabaras, N. (2018). Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. *Journal of Computational Physics*, *366*, 415–447.