

MULTICLASS CLASSIFICATION USING TRANSFER LEARNING COMBINED WITH CLASSICAL ML ALGORITHM

STEFANO BIONDI

1. TASK AND DATASET

Il task richiede l'utilizzo di una tecnica di transfer learning per la classificazione di immagini. Il Transfer Learning è un metodo di Machine Learning in cui un modello sviluppato e trained per un task è riutilizzato come input per un modello che risolve un secondo task. In particolare è richiesto di utilizzare una *Convolutional Neural Network* pretrained sul dataset IMAGENET. In questo elaborato verrà utilizzato l'output del modello VGG16 come input di due modelli di machine learning classici: il *Gradient Boosting for classification* e il *Linear Support Vector Classification*.

VGG16 è una rete neurale convoluzionale proposta da K. Simonyan and A. Zisserman che raggiunge il 92.7% di accuracy per i top 5 test set ricavati dal dataset ImageNet. Questo è un dataset di oltre 14 milioni di immagini suddivise in 1000 classi.

L'output della rete VGG16 è quindi formato da 1000 nodi. Il task affrontato in questo lavoro è invece un problema di classificazione a 5 classi:

- beaver
- brontosaurus
- camera
- cannon
- ceilingfan

Si sono quindi utilizzati alcuni blocchi delle rete neurale convoluzionale VGG16 come feature extraction del training e test set e, l'output della rete convoluzionale, è stato utilizzato come input dei due algoritmi di machine learning classico.

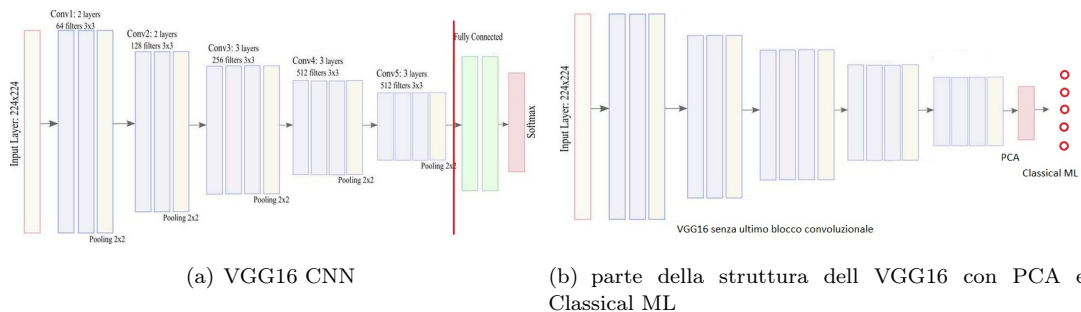


FIGURA 1. VGG16 e struttura del transfer learning utilizzando parte della CNN VGG16

Come training e test set è stata utilizzata una versione già splittata in training e test set del dataset *Caltech 101*. Essendo che per ogni classe sono state raccolte meno di 100 immagini non è stato creato il validation set, evitando quindi, un ulteriore split.

In fase di caricamento delle immagini è stato effettuato una resize con Antialiasing in formato 224x224x3 per permettere al modello VGG16 di utilizzare correttamente le immagini in input.

Infine il training e il test set sono stati mescolati internamente e le variabili target sono state categorizzate.

Come mostrato in figura (2) la distribuzione di frequenza delle classi è bilanciata, quindi nessuna tecnica di over o under sampling è stata applicata.

```
[[ 'beaver' 'brontosaurus' 'camera' 'cannon' 'ceiling_fan'
  '0.19827586206896552' '0.1896551724137931' '0.21551724137931033'
  '0.1896551724137931' '0.20689655172413793' ]]
```

FIGURA 2. Distribuzione di Frequenza Classe Target

2. MODELS

Si è deciso di tagliare rispettivamente 1, 6 e 10 strati del SVGG16 e utilizzare l'output come input dei due modelli di machine learning classici menzionati precedentemente.

Lo SVGG16 ha restituito rispettivamente 200.704, 100.352 e 512 parametri che, per gli algoritmi di machine learning classico, sono troppi. E' stata quindi effettuata una *Analisi delle Componenti Principali*, con una selezione del numero delle componenti tale che la varianza necessaria alla non perdita di informazioni fosse almeno pari al 95%. Questa riduzione della dimensionalità dell'input ha reso quindi più performante il training dei due algoritmi di classificazione, il Gradient Boosting for classification e il Linear Support Vector Classification.

Infine per valutare i modelli sono state utilizzate la *cross validation* e l'*accuracy* del test set. In figura 3 i risultati

```
Gradient Boosting Accuracy CV & variance: 0.8654079254079254 0.13767243137495377
Gradient Boosting accuracy test: 0.8849557522123894
Linear SVC Accuracy & variance: 0.9733333333333334 0.0832666399786453
SVC accuracy test: 0.9823008849557522
```

(a) SVGG16 senza ultimo strato

```
Gradient Boosting Accuracy CV & variance: 0.7164685314685315 0.1423900191219417
Gradient Boosting accuracy test: 0.7256637168141593
Linear SVC Accuracy & variance: 0.6678671328671328 0.1500447224966046
SVC accuracy test: 0.9203539823008849
```

(b) SVGG16 senza ultimi 6 strati

```
Gradient Boosting Accuracy CV & variance: 0.5214568764568763 0.15800607612966064
Gradient Boosting accuracy test: 0.6460176991150443
Linear SVC Accuracy & variance: 0.6154428904428905 0.19827027481203263
SVC accuracy test: 0.7964601769911505
```

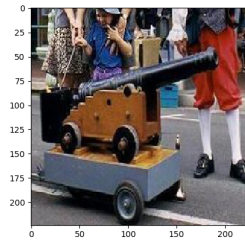
(c) SVGG16 senza ultimi 10 strati

FIGURA 3. Performance degli Ottimizzatori con parametri di default, Validation Set splittato dal Training Set

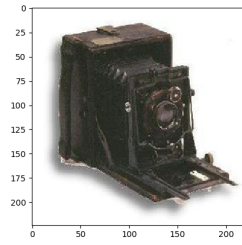
Come si può osservare entrambi i modelli di machine learning classico perdono di predittività con la riduzione degli strati dello VGG16. Questa notevole differenza è data dal fatto che eliminando gli ultimi blocchi di convoluzione il modello perde in riconoscimento dei dettagli delle immagini, passando da una media di accuracy nella cross validation dello 0.61 nel modello che utilizza solamente i primi 3 blocchi convoluzionali del VGG16 allo 0.97 del modello che utilizza come input l'output della rete con 5 blocchi convoluzionali del VGG16.

E' stata raggiunta un'accuracy del test set dello 0.98 senza la presenza di overfitting, infatti la media e la varianza delle accuracy della cross validation con *cv* pari a 10 risultano rispettivamente 0.97 e 0.08. Provando a modificare il Penalty parameter del termine di errore *c* dal valore di default, che è pari a 1, non ha migliorato le performance del modello. Ci riteniamo quindi soddisfatti nell'aver raggiunto un ottimo risultato predittivo utilizzando tutti e 5 i blocchi della CNN VGG16, PCA con mantenimento della varianza dell'informazione

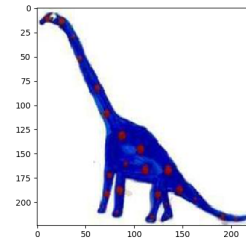
non inferiore al 95% e Support Vector Classification per la predizione della classe appartenente. In figura 4 le prime 5 immagini del test set con le rispettive predizioni.



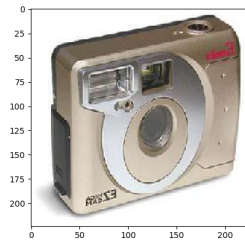
(a) Figure 1



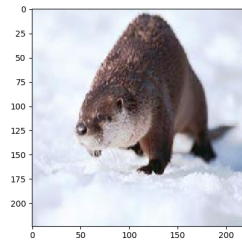
(b) Figure 2



(c) Figure 3



(d) Figure 4



(e) Figure 5

```
Predicted image: cannon
Predicted image: camera
Predicted image: brontosaurus
Predicted image: camera
Predicted image: beaver
```

(f) Predizioni

FIGURA 4. Prime 5 immagini del test set con rispettive predizioni