# Convolutional neural networks for classification of transmission electron microscopy imagery

Sergii Gryshkevych

Uppsala University

December 14, 2016

# Introduction

Introduction

# Performance measures

## The Accuracy Paradox

Models with a given accuracy may have greater predictive power than models with higher accuracy.

## Confusion matrix

| | Predicted True | Predicted False |
|---|---|---|
| Actual True | True Positive | False Negative |
| Actual False | False Positive | True Negative |

# Performance measures

## True positive rate (TPR)

AKA sensitivity or recall: $TPR = \frac{TP}{TP+FN}$

# Performance measures

## True positive rate (TPR)

AKA sensitivity or recall: $TPR = \frac{TP}{TP+FN}$

## True negative rate (TNR)

AKA specificity: $TNR = \frac{TN}{TN+FP}$

# Performance measures

## True positive rate (TPR)

AKA sensitivity or recall: $TPR = \frac{TP}{TP+FN}$

## True negative rate (TNR)

AKA specificity: $TNR = \frac{TN}{TN+FP}$

## Positive predicted value (PPV)

AKA precision: $PPV = \frac{TP}{TP+FP}$

# Performance measures

## True positive rate (TPR)

AKA sensitivity or recall: $TPR = \frac{TP}{TP+FN}$

## True negative rate (TNR)

AKA specificity: $TNR = \frac{TN}{TN+FP}$

## Positive predicted value (PPV)

AKA precision: $PPV = \frac{TP}{TP+FP}$

## Negative predicted value (NPV)

$NPV = \frac{TN}{TN+FN}$

# Performance measures

## True positive rate (TPR)

AKA sensitivity or recall: $TPR = \frac{TP}{TP+FN}$

## True negative rate (TNR)

AKA specificity: $TNR = \frac{TN}{TN+FP}$

## Positive predicted value (PPV)

AKA precision: $PPV = \frac{TP}{TP+FP}$

## Negative predicted value (NPV)

$NPV = \frac{TN}{TN+FN}$

## $F_1$ score

It is a harmonic mean of TPR and TNR

# Data augmentation techniques

- Rotation in the range $[-180, 180]$ degrees with spline interpolation

# Data augmentation techniques

- Rotation in the range $[-180, 180]$ degrees with spline interpolation
- Shear transformation in the range $[0, 0.2]$

# Data augmentation techniques

- Rotation in the range $[-180, 180]$ degrees with spline interpolation
- Shear transformation in the range $[0, 0.2]$
- Vertical shift in the range $[-10, 10]$ percent of total height

# Data augmentation techniques

- Rotation in the range $[-180, 180]$ degrees with spline interpolation
- Shear transformation in the range $[0, 0.2]$
- Vertical shift in the range $[-10, 10]$ percent of total height
- Horizontal shift in the range $[-10, 10]$ percent of total width

# Data augmentation techniques

- Rotation in the range $[-180, 180]$ degrees with spline interpolation
- Shear transformation in the range $[0, 0.2]$
- Vertical shift in the range $[-10, 10]$ percent of total height
- Horizontal shift in the range $[-10, 10]$ percent of total width
- Zoom in the range $[0.8, 1.0]$ which means zoom by a maximum 20%

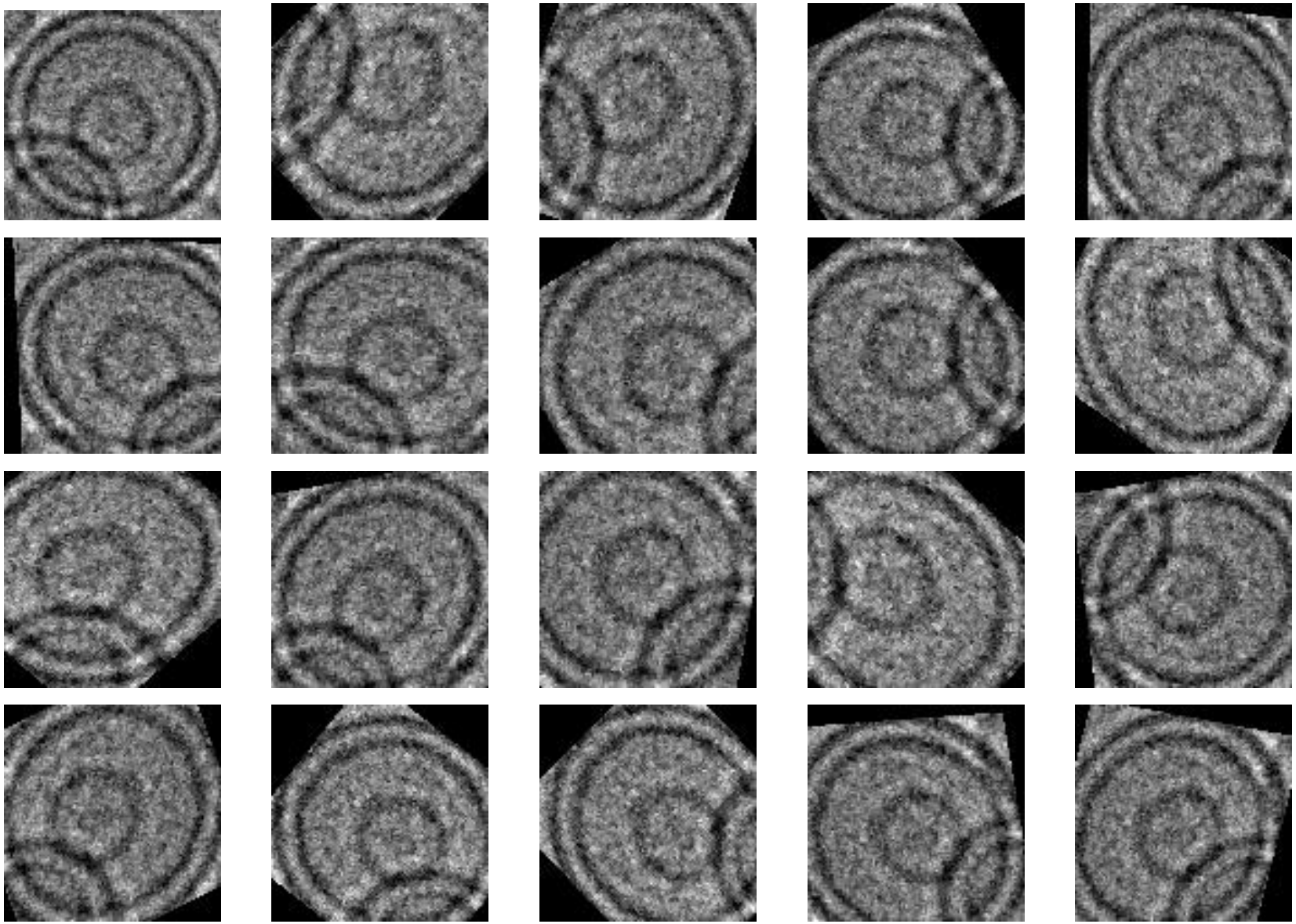
# Data augmentation techniques

- Rotation in the range $[-180, 180]$ degrees with spline interpolation
- Shear transformation in the range $[0, 0.2]$
- Vertical shift in the range $[-10, 10]$ percent of total height
- Horizontal shift in the range $[-10, 10]$ percent of total width
- Zoom in the range $[0.8, 1.0]$ which means zoom by a maximum 20%
- Horizontal flip

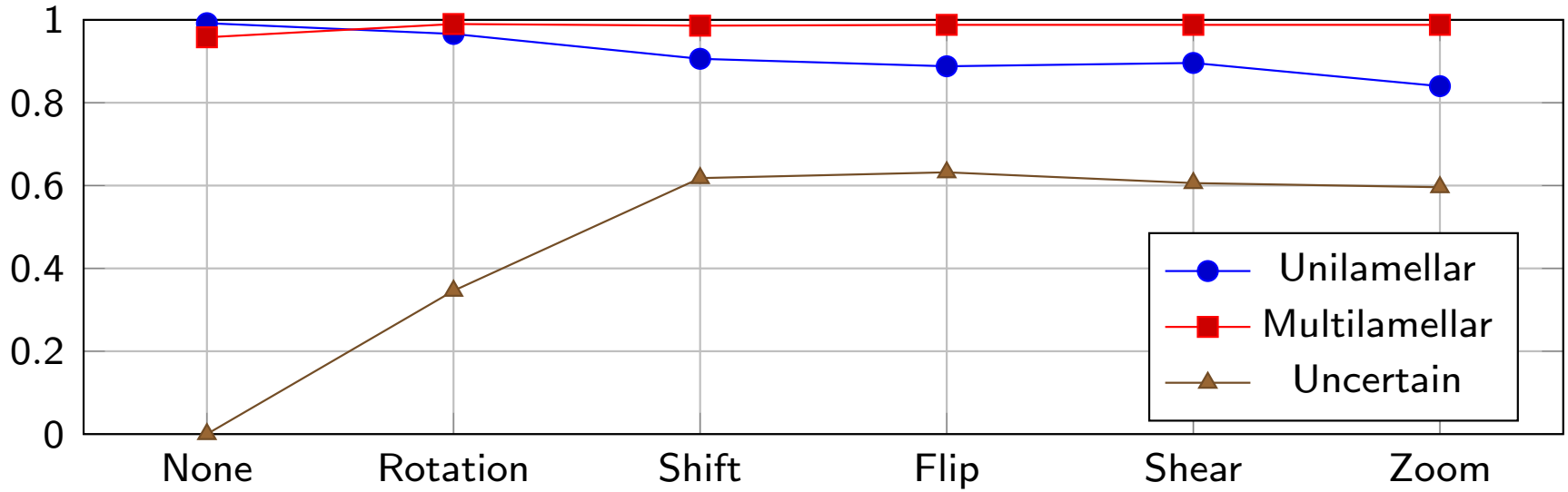
# Data augmentation techniques

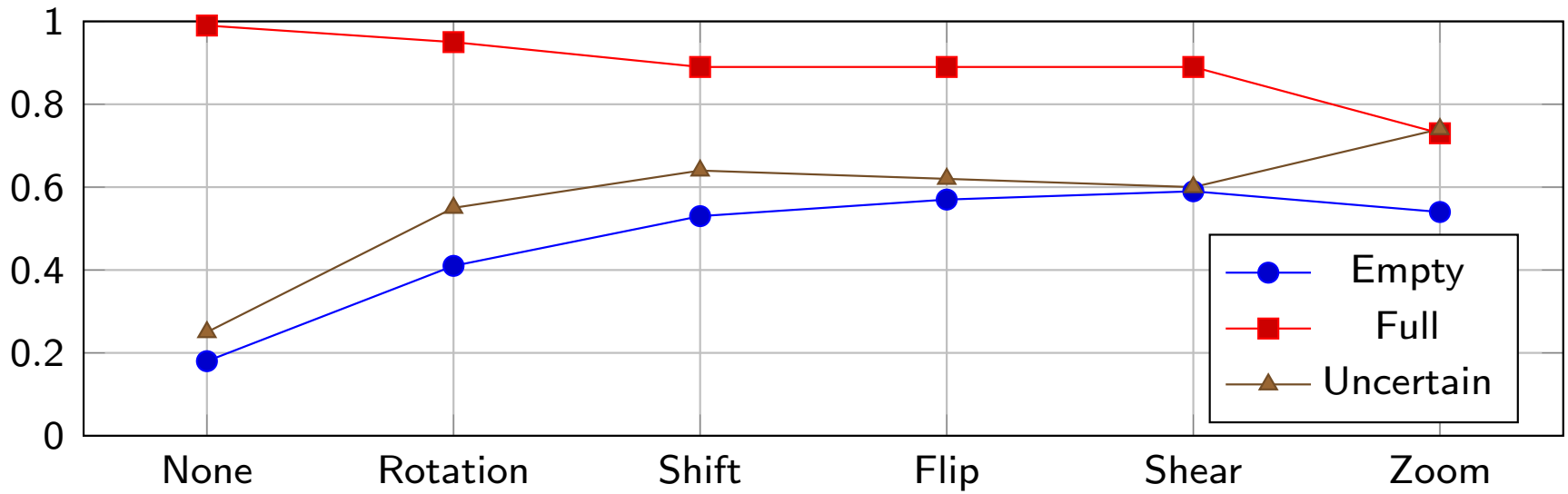
- Rotation in the range $[-180, 180]$ degrees with spline interpolation
- Shear transformation in the range $[0, 0.2]$
- Vertical shift in the range $[-10, 10]$ percent of total height
- Horizontal shift in the range $[-10, 10]$ percent of total width
- Zoom in the range $[0.8, 1.0]$ which means zoom by a maximum 20%
- Horizontal flip
- Vertical flip

# Data augmentation example

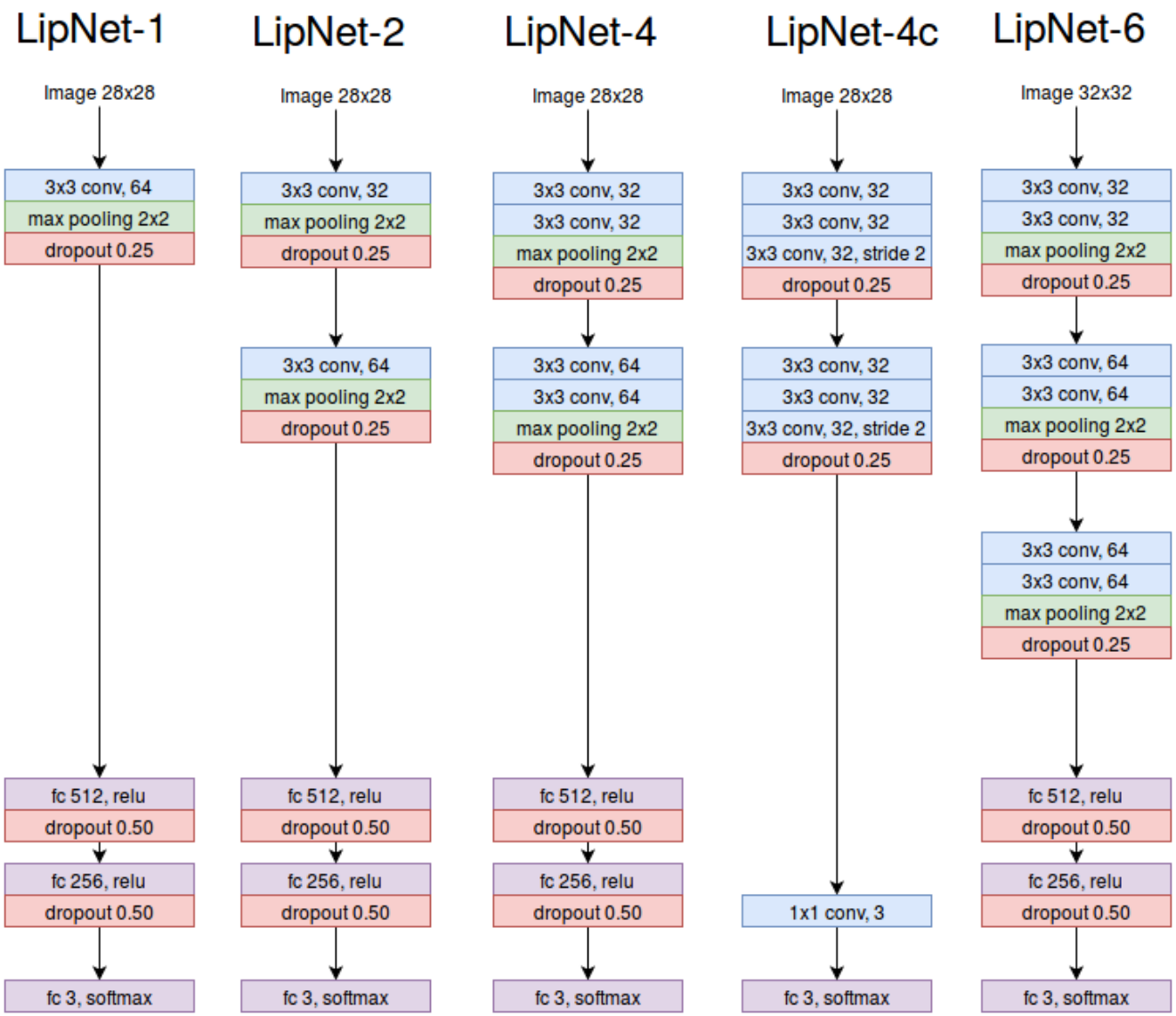


# Effect of the data augmentation

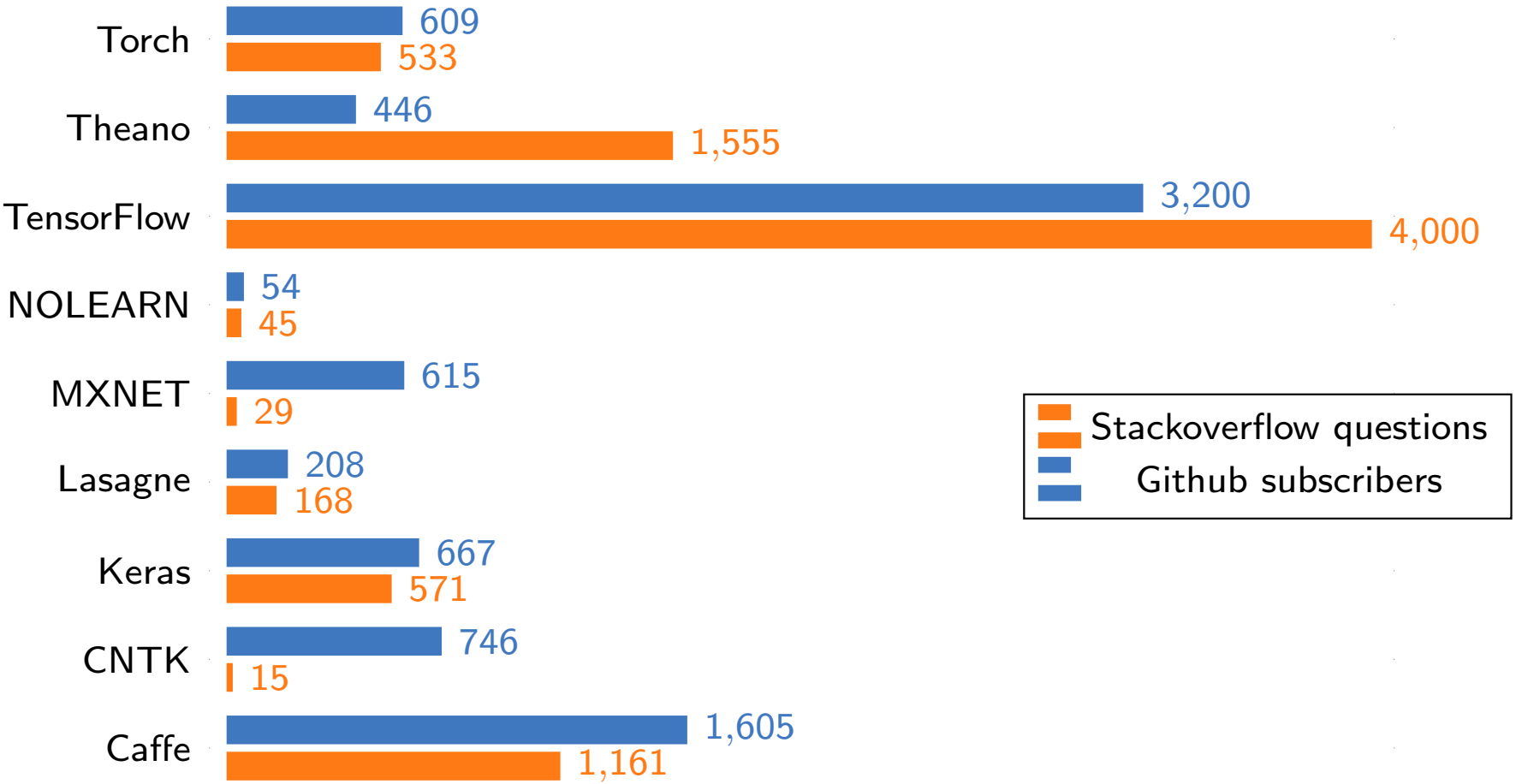


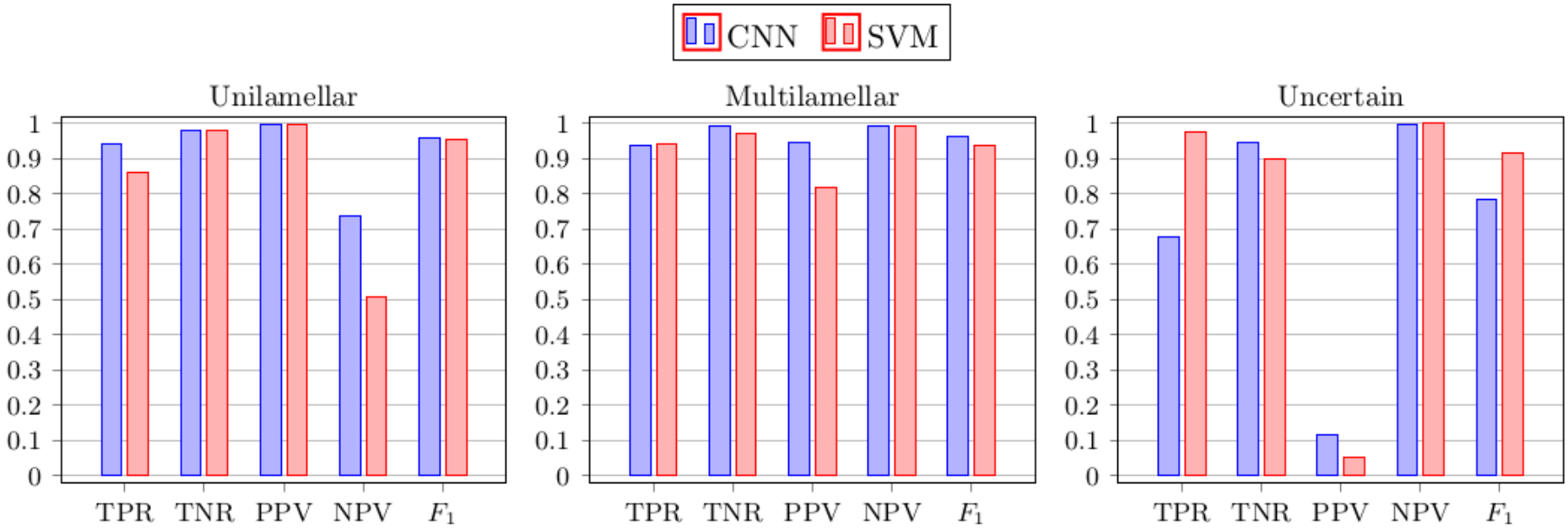
# Network architectures



# Popularity of deep learning software as of October 2016



# CNN vs SVM: Lamellarity



# CNN vs SVM: Encapsulation