

Source code

QUANTUM MACHINE LEARNING –BASED DETECTION OF FAKE NEWS AND DEEP FAKE VIDEOS, IMAGES

Python code for implementation

```
# Text Data Preprocessing for Fake News

import pandas as pd

from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder

# Load your dataset

df = pd.read_csv("fake_news_dataset.csv")

# Text preprocessing (e.g., removing stopwords, tokenization)
tfidf = TfidfVectorizer(stop_words='english', max_features=5000)
X_text = tfidf.fit_transform(df['text']) # Text features

# Encoding labels (Fake/Real)

le = LabelEncoder()
```

```
y_text = le.fit_transform(df['label']) # 0 for fake, 1 for real
```

```
# Split dataset into training and testing
```

```
X_train, X_test, y_train, y_test = train_test_split(X_text, y_text,  
test_size=0.3, random_state=42)
```

Python code for implementation

```
from sklearn.ensemble import StackingClassifier
```

```
from sklearn.svm import SVC
```

```
from sklearn.ensemble import RandomForestClassifier
```

```
from sklearn.linear_model import LogisticRegression
```

```
# Classical base models
```

```
base_learners = [
```

```
    ('rf', RandomForestClassifier(n_estimators=100)),
```

```
    ('svm', SVC(kernel='linear')),
```

```
    # You can add your quantum model as a base learner
```

```
    ('quantum_model', qsvm)
```

```
]
```

```
# Meta-learner
```

```
meta_learner = LogisticRegression()
```

```
# Stacked model
```

```
stacked_model = StackingClassifier(estimators=base_learners,  
final_estimator=meta_learner)
```

```
# Train the stacked model
```

```
stacked_model.fit(X_train, y_train)
```

```
# Evaluate
```

```
accuracy = stacked_model.score(X_test, y_test)
```

```
print(f"Accuracy of stacked model: {accuracy}")
```