Đánh giá hiệu quả của các phương pháp dự đoán giá cổ phiếu bằng các thuật toán học máy

1st Nguyễn Hữu Thắng

IS403.N21

Trường đại học Công Nghệ Thông Tin

[20520759@gm.uit.edu.vn](mailto:20520759@gm.uit.edu.vn)

2nd Trần Thu Thảo

IS403.N21

Trường đại học Công Nghệ Thông Tin

[20520769@gm.uit.edu.vn](mailto:20520769@gm.uit.edu.vn)

3rd Võ Thị Hà Trang

IS403.N21

Trường đại học Công Nghệ Thông Tin

[20522043@gm.uit.edu.vn](mailto:20522043@gm.uit.edu.vn)

4th Võ Phạm Thùy Nhung

IS403.N21

Trường đại học Công Nghệ Thông Tin

[20520679@gm.uit.edu.vn](mailto:20520679@gm.uit.edu.vn)

5th Nguyễn Đình Trải

IS403.N21

Trường đại học Công Nghệ Thông Tin

[19522371@gm.uit.edu.vn](mailto:19522371@gm.uit.edu.vn)

*Tóm tắt* — Bài báo này tập trung vào đánh giá hiệu quả của các phương pháp dự đoán giá cổ phiếu sử dụng các thuật toán học máy như ARIMA, Holt-Winters, TDNN, BVAR, NNAR, AAE, Linear Regression, RNN, LSTM và GRU. Dự đoán giá cổ phiếu là một thách thức quan trọng trong lĩnh vực tài chính, vì nó có vai trò quan trọng trong việc hỗ trợ quyết định đầu tư và quản lý rủi ro. Kết quả đánh giá hiệu quả của các phương pháp dự đoán giá cổ phiếu cung cấp thông tin quan trọng về độ chính xác và đáng tin cậy của dự đoán. Những phát hiện từ nghiên cứu này có thể hỗ trợ nhà đầu tư và các chuyên gia tài chính trong việc đưa ra quyết định đầu tư thông minh và phân tích thị trường tài chính. Hiểu rõ được hiệu quả của các thuật toán học máy trong dự đoán giá cổ phiếu có thể cung cấp lợi ích to lớn trong việc tối ưu hóa lợi nhuận và quản lý rủi ro trong lĩnh vực đầu tư chứng khoán.

Từ khóa — dự đoán giá cổ phiếu, thuật toán học máy, đầu tư, ARIMA, Holt-Winters, TDNN, BVAR, NNAR, AAE, Linear Regression, RNN, LSTM và GRU.

# Giới thiệu

Trong lĩnh vực đầu tư và quản lý tài chính, dự đoán giá cổ phiếu đóng vai trò quan trọng trong việc hỗ trợ quyết định đầu tư và quản lý rủi ro. Để đạt được sự thành công trong các hoạt động này, nhà đầu tư và chuyên gia tài chính cần có thông tin chính xác và đáng tin cậy về giá cổ phiếu. Do đó, việc nghiên cứu và đánh giá hiệu quả của các phương pháp dự đoán giá cổ phiếu trở thành một vấn đề quan trọng. Trong thời gian gần đây, các thuật toán học máy đã phát triển mạnh mẽ và trở thành công cụ quan trọng trong việc dự đoán giá cổ phiếu. Nhờ khả năng học từ dữ liệu lịch sử và xây dựng mô hình toán học và thống kê, các thuật toán này có thể cung cấp những dự đoán tự động và chính xác về giá cổ phiếu. Việc sử dụng các thuật toán học máy trong dự đoán giá cổ phiếu hứa hẹn mang lại nhiều lợi ích và tiềm năng trong việc quyết định đầu tư. Tuy nhiên, để áp dụng thành công các thuật toán học máy trong dự đoán giá cổ phiếu, chúng ta cần có sự so sánh và đánh giá giữa các phương pháp khác nhau. Điều này giúp chúng ta hiểu rõ hơn về hiệu quả và đáng tin cậy của từng thuật toán. Đồng thời, việc tìm ra những thuật toán tốt nhất và phù hợp nhất với mục tiêu đầu tư và quản lý rủi ro là một yếu tố quan trọng để đạt được kết quả tối ưu.

Trong nghiên cứu này, chúng tôi sử dụng một các thuật toán để dự đoán giá cổ phiếu, bao gồm Linear Regression, ARIMA, Holt-Winters, TDNN, BVAR, NNAR, AAE, RNN, LSTM và GRU. Mỗi thuật toán đều có những ưu điểm và đặc điểm riêng, từ việc sử dụng thông tin quá khứ đến việc xử lý mô hình phi tuyến tính và mạng nơ-ron.

Mục tiêu của nghiên cứu này là đánh giá hiệu quả của các phương pháp dự đoán giá cổ phiếu sử dụng các thuật toán nêu trên. Chúng tôi sẽ sử dụng các chỉ số sai số như Mean Absolute Error (MAE), Mean Squared Error (MSE) và Root Mean Squared Error (RMSE) để đánh giá hiệu quả của từng phương pháp. Các kết quả đánh giá sẽ giúp chúng ta hiểu rõ hơn về khả năng và giới hạn của các thuật toán học máy trong việc dự đoán giá cổ phiếu. Chúng ta có thể xác định được những thuật toán nào mang lại kết quả tốt nhất và phù hợp nhất với mục tiêu đầu tư và quản lý rủi ro. Và thông qua việc tìm hiểu hiệu quả của các phương pháp dự đoán giá cổ phiếu, chúng ta có thể nâng cao khả năng tối ưu hóa lợi nhuận và quản lý rủi ro trong lĩnh vực đầu tư chứng khoán. Điều này có ý nghĩa quan trọng trong việc hỗ trợ quyết định đầu tư thông minh và đạt được hiệu suất cao trong môi trường thị trường tài chính..

# Nghiên cứu liên quan

# Kiến trúc mô hình

## Linear Regression

Trong trường hợp đơn giản nhất, mô hình hồi quy cho phép tồn tại mối quan hệ tuyến tính giữa biến phụ thuộc y và một biến độc lập duy nhất x.

Yt=β0+β1xt+εt

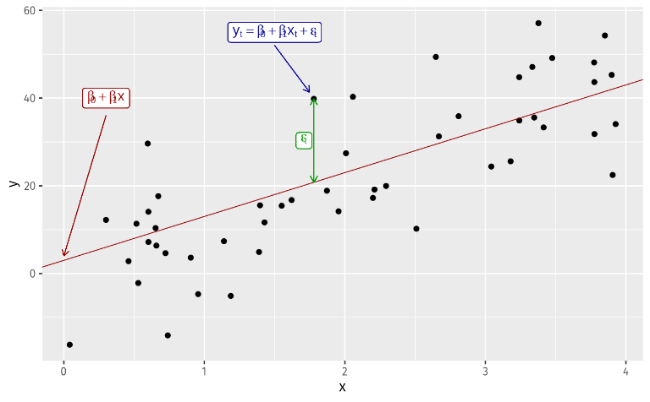
Trong đó:

Yt: là biến phụ thuộc

β0: hệ số chặn

β1: hệ số hồi quy cho biến độc lập

εt: sai số



Theo hình trên ta thấy các hệ số β0 và β1 lần lượt biểu thị giao điểm và hệ số góc của đường thẳng. Hệ số chặn β­0 đại diện cho giá trị dự đoán của y khi x=0, β1 đại diện cho sự thay đổi của y khi x thay đổi.

## Holt-Winters

Thuật toán Holt-winters là một phương pháp dự báo chuỗi thời gian, kết hợp Exponential Smoothing với các yếu tố mùa vụ (seasonality) và xu hướng (trend). Ba thành phần làm mịn (smoothing) cần thiết để dự báo trong mô hình holt-winters là level, trend, seasonal, những thành phần này nằm trong khoảng từ 0 đến 1.

## ARIMA

Mô hình ARIMA là viết tắt của quá trình tự hồi quy (AutoRegression -AR), quá trình trung bình trượt (Moving Average – MA) và tích hợp sai phân Integrated – I.

* *Các loại mô hình ARIMA*

Mô hình ARIMA không có tính mùa vụ

Mô hình ARIMA có tính mùa vụ (Seasonal ARIMA – SARIMA)

* *Chuỗi dừng*

Một chuỗi thời gian có tính dừng là một chuỗi các giá trị mean, variance, autocorrelation không thay đổi theo thời gian và nó không bao hàm các yếu tố xu thế. Với hầu hết các phương pháp thống kê dự báo, đều phải đảm bảo tính dừng của chuỗi dữ liệu vì thế việc kiểm tra tính dừng là rất quan trọng.

Để kiểm định tính dừng của dữ liệu ta có hai phương pháp kiểm định phổ biến: Kiểm định Dickey Fuller3(DF) và Dickey Fuller cải tiến(ADF4).

* *Mô hình ARIMA không có tính mùa vụ*

Với p,d,q lần lượt là các số không âm

I(d): Integrated – So sánh sự khác nhau giữa d quan sát (Hiệu giữa giá trị hiện tại và d giá trị trước đó).

AR(p): Autoregression – là quá trình tìm mối quan hệ giữa dữ liệu hiện tài p dữ liệu quá khứ trước đó. (Gọi là lag)

yt = a0 + a1 yt-1 + a­2 yt-2 + … + ap yt-p + ԑt

Điều kiện dừng của việc chọn p:

MA(q): Moving Average – là quá trình tìm mối qua hệ giữa dữ liệu hiện tại và q phần lỗi quá khứ trước đó

yt =β0 + β1 ԑt-1 + β2 ԑt-2 + … + βq ԑt-q + µt

Điều kiện dừng của việc chọn q:

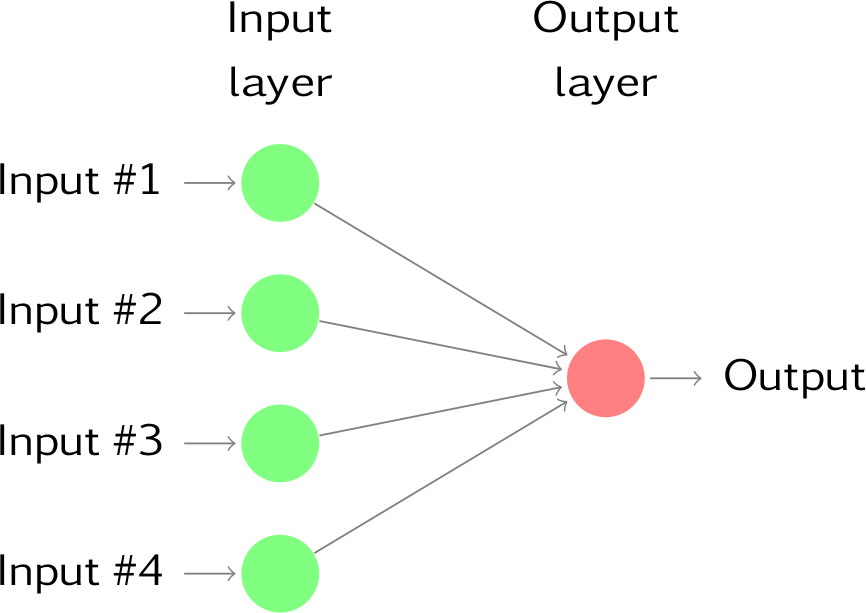
## TDNN

TDNN (Time Delay Neural Network) là một mô hình mạng nơ-ron thời gian được sử dụng để mô hình hóa và dự đoán chuỗi thời gian. Nó khác biệt với mạng nơ-ron thông thường bằng cách kết nối các nút ẩn với các giá trị vào của các bước thời gian trước đó. Cấu trúc này cho phép TDNN nhìn vào quá khứ và sử dụng thông tin lịch sử để dự đoán giá trị tiếp theo trong chuỗi thời gian. Công thức tổng quát cho một nút ẩn trong TDNN có thể được biểu diễn như sau:

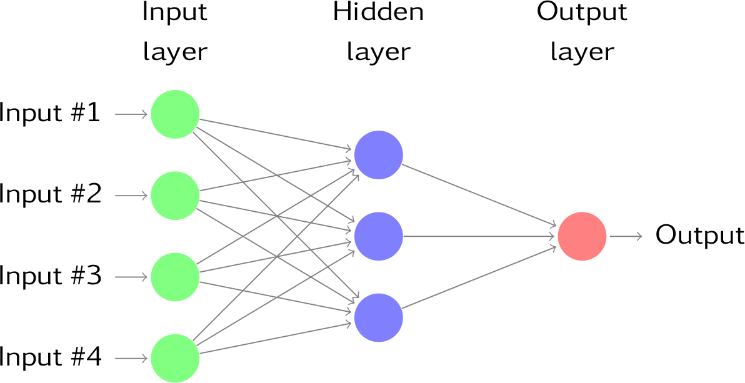
## NNAR

* *Kiến trúc mạng nơ-ron (Neural Network)*

Mạng nơ-ron bao gồm nhiều nơ-ron được tổ chức thành các lớp. Các lớp bao gồm lớp đầu vào, lớp đầu ra và cũng có thể có lớp trung gian chứa các nơ-ron ẩn. Với một mạng nơ-ron đơn giản nhất sẽ không chứa các lớp ẩn và lúc này nó tương đương với hồi quy tuyến tính. Dưới đây là hình minh họa một mạng nơ-ron đơn giản:



Khi chúng ta thêm một lớp trung gian với các nơ-ron ẩn, mạng nơ-ron sẽ trở nên phi tuyến tính được minh họa như hình bên dưới:



* *Neural Network Autoregression* (*NNAR)*

Tương tự như khi sử dụng các giá trị lag trong mô hình hồi quy tuyến tính, khi dùng các quan sát trong quá khứ (các giá trị lag) làm đầu vào cho mạng nơ-ron để dự đoán các giá trị đầu ra ta gọi đây là mô hình mạng nơ-ron tự hồi quy - Neural Network Autoregression (NNAR). Xét các mạng chuyển tiếp với một lớp ẩn được ký hiệu:

NNAR(P, k), trong đó P là số lag đầu vào, k là các nút trong lớp ẩn.

* *Hàm tương quan một phần (PACF)*

PACF hay còn gọi là hàm tự tương quan một phần là hàm dùng để xác định bậc tự tương quan trên laq đến p. PACF này được viết bằng phương trình:

## RNN

A picture containing text, font, screenshot, line

Description automatically generatedRNN (Recurrent Neural Network) là một loại neural network được sử dụng chủ yếu cho dữ liệu có cấu trúc chuỗi như chuỗi thời gian, ngôn ngữ tự nhiên và dữ liệu văn bản. Thuật toán RNN sử dụng khái niệm "trạng thái ẩn" (hidden state) để duy trì thông tin về quá khứ và sử dụng thông tin đó khi xử lý dữ liệu trong tương lai.

Chúng ta ký hiệu trạng thái ẩn và đầu vào tại thời điểm t lần lượt là Ht ∈ Rn x h và Xt∈Rn x d, trong đó n là số lượng mẫu, d là số lượng đầu vào của mỗi mẫu và h là số lượng đơn vị ẩn. Hơn nữa, chúng ta sử dụng ma trận trọng số Wxh ∈ Rd×h, ma trận Whh ∈ Rh×h để truyền thông tin từ trạng thái ẩn đến trạng thái ẩn, và tham số bias bh ∈ R1×h.

Cuối cùng, tất cả thông tin này được truyền qua một hàm kích hoạt φ, thường là hàm sigmoid logistic hoặc hàm tanh, để chuẩn bị các gradient để sử dụng trong quá trình lan truyền ngược. Kết hợp tất cả các ký hiệu này lại với nhau, ta thu được Phương trình 1 là biến ẩn ẩn và Phương trình 2 là biến đầu ra.

Ht = φh (XtWxh + Ht−1Whh + bh) (1)

Ot = φo (HtWho + bo) (2)

Vì Ht được tính toán đệ quy dựa trên Ht−1 và quá trình này diễn ra cho mỗi bước thời gian, RNN bao gồm các dấu vết của tất cả các trạng thái ẩn trước đó của Ht−1 cũng như chính Ht−1.

Nếu chúng ta so sánh notation cho RNN với notation tương tự cho Feedforward Neural Networks thẳng, chúng ta có thể thấy rõ sự khác biệt đã được mô tả trước đó. Trong Phương trình 3, chúng ta có thể thấy quá trình tính toán cho biến ẩn ẩn, trong khi Phương trình 4 thể hiện biến đầu ra.

H = φh (XWxh + bh) (3)

O = φo (HWho + bo) (4)

RNN được sử dụng rộng rãi trong nhiều lĩnh vực như xử lý ngôn ngữ tự nhiên, dịch máy, nhận dạng giọng nói, dự đoán chuỗi thời gian và nhiều ứng dụng khác có liên quan đến dữ liệu chuỗi.

## LTSM

LSTM là một loại RNN đặc biệt với các tính năng bổ sung để ghi nhớ chuỗi dữ liệu. Việc ghi nhớ xu hướng trước đó của dữ liệu có thể thực hiện được thông qua một số cổng cùng với một dòng bộ nhớ được tích hợp trong một LSTM điển hình.

Mỗi LSTM là một tập hợp các ô hoặc mô-đun hệ thống, nơi các luồng dữ liệu được thu thập và lưu trữ. Các ô giống như một đường vận chuyển (đường phía trên trong mỗi ô) kết nối từ mô-đun này sang mô-đun khác truyền dữ liệu từ quá khứ và thu thập chúng cho mô-đun hiện tại. Do việc sử dụng một số cổng trong mỗi ô, dữ liệu trong mỗi ô có thể được xử lý, lọc hoặc thêm vào cho các ô tiếp theo.

Cấu trúc bên trong của một ô LSTM được thể hiện trong hình:

A picture containing diagram, line, plan, technical drawing

Description automatically generated

Một mô hình LSTM bao gồm ba cổng: *cổng quên* (forget gate), *cổng đầu vào* (input gate) và *cổng đầu ra* (output gate).

* *Forget gate:* Hàm sigmoid thường được sử dụng cho cổng này để đưa ra quyết định về thông tin nào cần được xóa khỏi bộ nhớ LSTM. Đầu ra của cổng này là ft, có giá trị nằm trong khoảng từ 0 đến 1, trong đó 0 biểu thị loại bỏ hoàn toàn giá trị đã học và 1 ngụ ý bảo toàn toàn bộ giá trị. Đầu ra này được tính như sau:

*ft = (Wf [ht - 1], Wf [xt], bf)*

* *Input gate*: Cổng này đưa ra quyết định liệu thông tin mới có được thêm vào bộ nhớ LSTM hay không. Cổng này bao gồm hai lớp: 1) lớp *sigmoid* và 2) lớp “*tanh*”. Lớp *sigmoid* quyết định những giá trị nào cần được cập nhật và lớp *tanh* tạo một vectơ chứa các giá trị ứng cử viên mới sẽ được thêm vào bộ nhớ LSTM. Đầu ra của hai lớp này được tính toán như sau:

*it = (Wi[ht-1], Wi[xt], bi)*

*c˜t = tanh(Wc[ht-1], Wc[xt], bc)*

Sự kết hợp của hai lớp này cung cấp một bản cập nhật cho bộ nhớ LSTM trong đó giá trị hiện tại bị quên bằng cách sử dụng lớp cổng quên thông qua phép nhân giá trị cũ ( tức là *ct-1*), sau đó thêm giá trị ứng cử viên mới vào *it\* c˜t* . Phương trình sau biểu diễn phương trình toán học của nó:

*ct = ft \* ct-1 + it\* c˜t*

* *Output gate:* Đầu tiên, cổng này sử dụng một lớp *sigmoid* để đưa ra quyết định phần nào của bộ nhớ LSTM đóng góp vào đầu ra. Sau đó, nó thực hiện một hàm *tanh* phi tuyến tính để ánh xạ các giá trị giữa −1 và 1. Cuối cùng, kết quả được nhân với đầu ra của một lớp sigmoid. Phương trình sau đây biểu thị các công thức để tính toán đầu ra:

*ot = (Wo[ht-1], Wo[xt], bo)*

*ht = ot \* tanh(ct)*

trong đó *ot* là giá trị đầu ra và *ht* là biểu diễn của nó dưới dạng giá trị giữa −1 và 1.

## GRU

Gated Recurrent Unit (GRU) là một loại mô hình mạng nơ-ron hồi quy (RNN) được giới thiệu nhằm giải quyết vấn đề mất mát thông tin xa trong các mạng RNN truyền thống. GRU sử dụng cơ chế cổng để điều chỉnh quá trình truyền thông tin trong mạng và giúp mô hình nhớ được thông tin quan trọng trong quá khứ.

Ảnh có chứa ảnh chụp màn hình, biểu đồ, hình vuông

Mô tả được tạo tự động

GRU có cấu trúc đơn giản hơn so với mô hình Long Short-Term Memory (LSTM) nhưng vẫn đạt được hiệu suất tương tự trong nhiều nhiệm vụ. Nó bao gồm hai cổng chính:

* Update gate: Cổng cập nhật xác định tỷ lệ thông tin nào sẽ được cập nhật từ trạng thái trước đó và thông tin mới nào sẽ được truyền vào trạng thái hiện tại. Nó quyết định liệu mô hình có nên quên hoặc ghi nhớ thông tin từ quá khứ.

Trong đó:

zt là giá trị của Update gate tại thời điểm t.

ht-1 là trạng thái ẩn trước đó.

xt là trạng thái đầu vào hiện tại.

Wz là ma trận trọng số cho Update gate.

* Reset gate: Cổng khôi phục xác định tỷ lệ thông tin nào từ trạng thái trước đó sẽ bị đặt lại (reset) để tạo không gian cho thông tin mới. Nó giúp loại bỏ những thông tin không cần thiết hoặc không quan trọng.

Trong đó:

rt là giá trị của Reset gate tại thời điểm t.

ht-1 là trạng thái ẩn trước đó.

xt là trạng thái đầu vào hiện tại.

Wr là ma trận trọng số cho Reset gate.

Các cổng cập nhật và khôi phục được kết hợp để tính toán trạng thái ẩn mới (ht) trong GRU. Các giá trị của hai cổng này sẽ quyết định phần nào của thông tin từ trạng thái trước đó sẽ được cập nhật và phần nào sẽ bị đặt lại, từ đó giúp mô hình hiểu và ghi nhớ thông tin quan trọng trong chuỗi dữ liệu.

GRU sử dụng một lớp ẩn duy nhất để tính toán trạng thái ẩn mới. Điều này giúp mô hình tiết kiệm tài nguyên tính toán và thời gian huấn luyện.

Tóm lại, GRU là một kiến trúc mạng nơ-ron hồi quy mạnh mẽ và phổ biến trong việc xử lý dữ liệu chuỗi, bao gồm xử lý ngôn ngữ tự nhiên, dịch máy, nhận dạng giọng nói và nhiều ứng dụng khác. Nó cải thiện khả năng mô hình ghi nhớ thông tin xa và giúp giải quyết vấn đề mất mát thông tin trong các mạng RNN truyền thống.

## BVAR

Bayesian Vector Autoregression (BVAR) sử dụng các phương pháp Bayes để ước lượng mô hình VAR. BVAR khác với các mô hình VAR tiêu chuẩn ở chỗ các tham số mô hình được coi là biến ngẫu nhiên, có xác suất biết trước, thay vì các giá trị cố định.

## AAE

AAE (Adversarial Autoencoder) là một kiến trúc mô hình học sâu kết hợp giữa **Autoencoder** và mạng **Adversarial** nhằm chủ yếu để tạo ra dữ liệu mới có cùng phân phối với dữ liệu huấn luyện ban đầu và thực hiện các nhiệm vụ như phân loại, phân đoạn và tái tạo dữ liệu.

A picture containing text, diagram, font, screenshot

Description automatically generated

* **Autoencoder** là một kiến trúc neural network được sử dụng để học biểu diễn tự động (latent representation) của dữ liệu đầu vào. Nó bao gồm hai phần chính: encoder (bộ mã hóa) và decoder (bộ giải mã). Encoder chuyển đổi dữ liệu đầu vào thành biểu diễn ẩn (latent representation) có chiều thấp hơn, trong khi decoder chuyển đổi biểu diễn ẩn trở lại dữ liệu ban đầu.
* Mạng **Adversarial** là một mạng được sử dụng để tạo ra dữ liệu giả mô phỏng dữ liệu thực tế. Nó bao gồm hai thành phần chính: generator (bộ sinh) và discriminator (bộ phân biệt). Generator nhận đầu vào là một biểu diễn ẩn ngẫu nhiên và cố gắng tạo ra dữ liệu giả mô phỏng dữ liệu thực tế. Trong khi đó, discriminator nhận đầu vào là dữ liệu thực tế và dữ liệu giả và cố gắng phân biệt chúng.

# Thực nghiệm

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll down window on the left of the MS Word Formatting toolbar.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **ARE** | **ARI** | **KREF** |
| **Count** | 2118 | 2118 | 2118 |
| **Mean** | 136.72024 | 15.47944 | 122.57269 |
| **Min** | 71.65 | 4.24 | 71.15 |
| **Max** | 223.57 | 19.68 | 229.44 |
| **25%** | 112.1875 | 13.05 | 96.32 |
| **50%** | 128.58 | 16.64 | 107.905 |
| **75%** | 160.8025 | 18.26 | 146.865 |
| **Box Plot** | Ảnh có chứa biểu đồ, ảnh chụp màn hình, hàng, Hình chữ nhật  Mô tả được tạo tự động |  | Ảnh có chứa biểu đồ, văn bản, ảnh chụp màn hình, hàng  Mô tả được tạo tự động |
| **Histogram** | Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, Sơ đồ  Mô tả được tạo tự động | Ảnh có chứa văn bản, biểu đồ, ảnh chụp màn hình, Sơ đồ  Mô tả được tạo tự động | Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, Sơ đồ  Mô tả được tạo tự động |
| **Nhận xét** |  |  |  |

## Authors and Affiliations

**The template is designed for, but not limited to, six authors.** A minimum of one author is required for all conference articles. Author names should be listed starting from left to right and then moving down to the next line. This is the author sequence that will be used in future citations and by indexing services. Names should not be listed in columns nor group by affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization).

### For papers with more than six authors: Add author names horizontally, moving to a third row if needed for more than 8 authors.

### For papers with less than six authors: To change the default, adjust the template as follows.

#### Selection: Highlight all author and affiliation lines.

#### Change number of columns: Select the Columns icon from the MS Word Standard toolbar and then select the correct number of columns from the selection palette.

#### Deletion: Delete the author and affiliation lines for the extra authors.

## Identify the Headings

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Component heads identify the different components of your paper and are not topically subordinate to each other. Examples include Acknowledgments and References and, for these, the correct style to use is “Heading 5”. Use “figure caption” for your Figure captions, and “table head” for your table title. Run-in heads, such as “Abstract”, will require you to apply a style (in this case, italic) in addition to the style provided by the drop down menu to differentiate the head from the text.

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced. Styles named “Heading 1”, “Heading 2”, “Heading 3”, and “Heading 4” are prescribed.

## Figures and Tables

#### Positioning Figures and Tables: Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation “Fig. 1”, even at the beginning of a sentence.

1. Table Type Styles

| Table Head | Table Column Head | | |
| --- | --- | --- | --- |
| Table column subhead | Subhead | Subhead |
| copy | More table copya |  |  |

1. Sample of a Table footnote. (*Table footnote*)

We suggest that you use a text box to insert a graphic (which is ideally a 300 dpi TIFF or EPS file, with all fonts embedded) because, in an MSW document, this method is somewhat more stable than directly inserting a picture.

To have non-visible rules on your frame, use the MSWord “Format” pull-down menu, select Text Box > Colors and Lines to choose No Fill and No Line.

1. Example of a figure caption. (*figure caption*)

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when writing Figure axis labels to avoid confusing the reader. As an example, write the quantity “Magnetization”, or “Magnetization, M”, not just “M”. If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write “Magnetization (A/m)” or “Magnetization {A[m(1)]}”, not just “A/m”. Do not label axes with a ratio of quantities and units. For example, write “Temperature (K)”, not “Temperature/K”.

##### Acknowledgment *(Heading 5)*

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

##### References

The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use “Ref. [3]” or “reference [3]” except at the beginning of a sentence: “Reference [3] was the first ...”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” [4]. Papers that have been accepted for publication should be cited as “in press” [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

1. G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*

**IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.**