

# Learning Meters of Arabic and English poems

With Recurrent Neural Networks

---

Prof. Waleed A. YOUSEF

The Team

July 17, 2018

Computer Science department

Faulty of Computers and Information, Helwan University

# Table of contents

1. Introduction
2. Literature Review
3. Deep Learning! (Our new approach)
4. Results
5. Details of Implementation and Results

# Introduction

---

فَقُولُ رَسُولِ اللَّهِ أَزْكَى وَأَشْرَحُ

وَدَعْ عَنْكَ آرَاءَ الرِّجَالِ وَقَوْلَهُمْ

# But ... What is poetry?

## General Definition:

- **Poetry** is a piece of writing or speaking, which **MUST** follow specific **Patterns**.

## Example, *English verse*:

That **time** of **year** thou **mayst** in **me** behold

To detect poems' meters, we need to learn those **Patterns**.

ودع عنك آراء الرجال وقولهم  
فقول رسول الله أذكى وأشرح

- A **poem** is a collection of verses.
- **Vowels** carry one of َ ُ ِ.
- **Consonants** carry ْ.
- A **foot** التفعيلة: is an **ordered** sequence of vowels and consonants.

Feet	Scansion
فَعُولُنْ	0/0//
فَاعِلُنْ	0//0/
مُسْتَفْعِلُنْ	0//0/0/
مَفَاعِيلُنْ	0/0/0//
مَفْعُولَات	0//0///
فَاعِلَاتُنْ	0/0//0/
مُفَاعِلَتُنْ	0///0//
مُتَفَاعِلُنْ	0//0///

- **Meter** البحر: is an **ordered** sequence of **feet**.

ويَسْأَلُ فِي الْحَوَادِثِ ذُو صَوَابٍ  
 وَيَسْأَلُ فُلَّ حَوَادِثِ ذُو صَوَابِينَ  
 0/0//      0///0//      0///0//  
 مَفَاعِلَتُنْ      مَفَاعِلَتُنْ      مَفَاعِلَتُنْ

Meter Name	Meter feet combination
<i>al-Wafeer</i>	مَفَاعِلَتُنْ مَفَاعِلَتُنْ فَعُولُنْ
<i>al-Taweel</i>	فَعُولُنْ مَفَاعِلَتُنْ فَعُولُنْ مَفَاعِلَتُنْ
⋮	⋮
<i>al-Moktadib</i>	مَفْعُولَاتُ مُسْتَفْعِلُنْ مُسْتَفْعِلُنْ
<i>al-Modar'e</i>	مَفَاعِلَتُنْ فَاعِلَاتُنْ مَفَاعِلَتُنْ

# English Prosody

## English Meters Building Blocks:

- **Syllables:**  $/^{\prime}wɔ:tə/ = /^{\prime}wɔ:/ + /tə(r)/$ .
  - **stressed** + unstressed.
- **Foot:** is a combination of stressed and unstressed syllables.

Feet	Stresses Combination
<i>Iamb</i>	$\times /$
<i>Trochee</i>	$/ \times$
<i>Dactyl</i>	$/ \times \times$
<i>Anapest</i>	$\times \times /$
<i>Pyrrhic</i>	$\times \times$
<i>Amphibrach</i>	$\times / \times$
<i>Spondee</i>	$//$

**Meter:** is repeating a foot  $n$  times; where  $n \in [1, 8]$ .



*Iambic pentameter* verse:

That **time** of **year** thou **mayst** in **me** be**hold**.  
Iambic Foot      2nd      3rd      4th      5th

# Literature Review

---

## Abuata and Al-Omari:

- Five-step Algorithm
  1. Getting the input, carrying full diacritics.
  2. Metrical scansion rules are applied to the Arud writing. 0/0/..
  3. Grouping zero and ones to feet **تفعيلات**.
  4. A class is assigned to the input.
- **Results:** 82.2% of 417 verses.

**Alnagdawi et al**, similar approach; Context-Free Grammar; 75% correctly classed from 128.

- Probabilistic operation

ويسأل في الحوادث ذو صوابٍ		
صوابين	حوادث ذو	ويسأل فل
0/0//	0///0//	0///0//
فعولن	مفاعلتن	مفاعلتن

Issues;

- A huge constrain. **Diacritics** are a must.
- Converting the text into pronounced text is **probabilistic**.
  - اثبات الحروف المحذوفة خطأً
  - التصرف في التقاء الساكنين

## Metric or Free-Verse:

- Verses are represented as vectors of **statistical features**.
  - Average number of feet per line.
  - Longest run of a single foot.
  - Percentage of foot changes.
- He has used an stress-annotated dataset, which means that the pattern is already detected!
- A couples of tries for detecting poem inside documents depending on visual features, for both Arabic and English.
  - Average of line length.
  - Average number of block.
  - etc ...

# Our point of departure

- Detecting poems inside documents by its visual feature represent the majority of the research related to poetry.
- For detecting meters, all models are so naive and primitive. They do not have any clue about the real pattern.

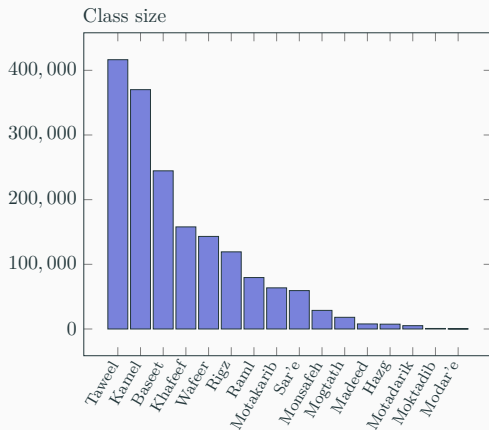
# Datasets

---



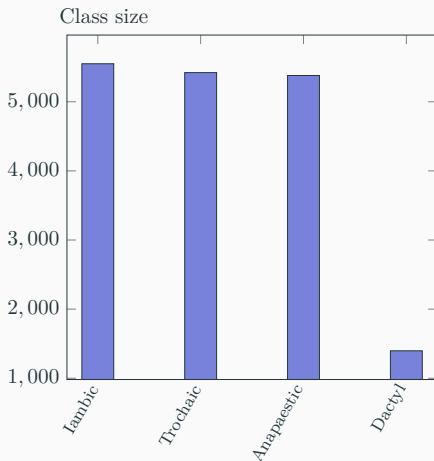
## Arabic Dataset:

- Our dataset consists of 1,722,321 data points labeled by the 16 meters.
- Verses are for 3701 poets from 11 ages.



## English Dataset:

- Our dataset consists of 17,744 verses only, labeled by 4 meters.
- This is what we have found after an extensive search.



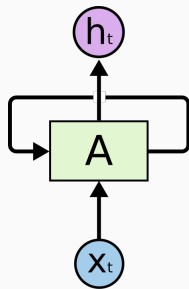
- The effect of the diacritics over the learning curve.
- The effect of the dataset size over the learning curve.
- Best technique to represent character-level text.

# Methodology

---

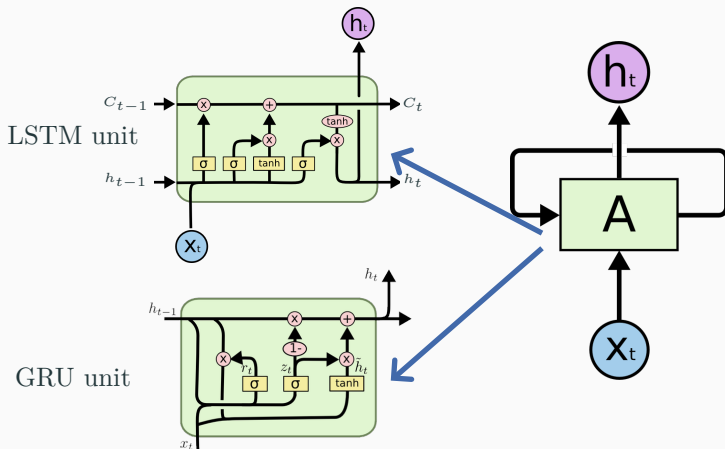
# Which Network!

- **Pattern:** is a sequence of characters.
- Unlike feedforward neural networks, RNNs can use their internal state (memory) to process sequences of inputs.



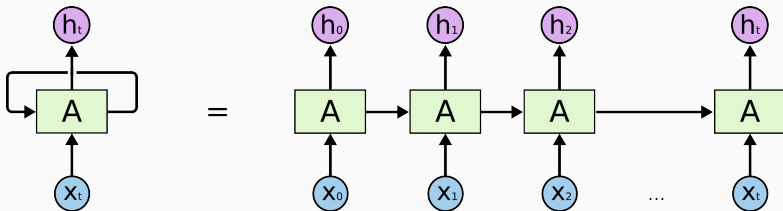
Rolled Rnn unit

# RNN, Architectures

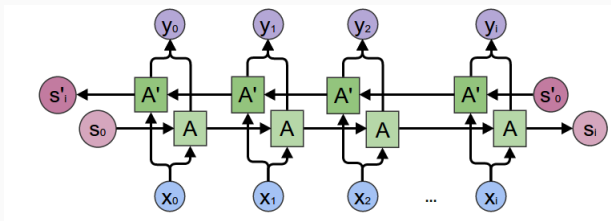


- Two variants of *recurrent units*.
- LSTM and GRU are designed for the same reason, to avoid vanishing problem and long term dependency problem .

# RNN, Architectures



Unidirectional RNN



Bidirectional RNN

An Issue:

- Diacritics are standalone characters!
  - $\text{len مرحبا} \neq \text{len مَرَحَبًا}$
  - We have represented the letter and its diacritic as a **one character**.

**Benefits:**

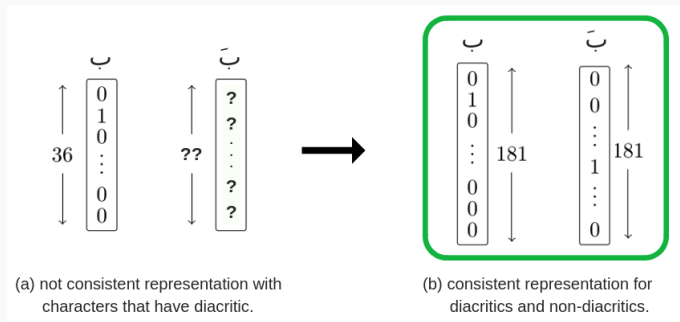
1. Verse's length is fixed, regardless the diacritic states.
2. Saving more space, by shorten the length of full diacritic verses.
3. Models can be tested on both diacritic or non-diacritic data.



# Encoding Techniques

1. One-Hot
2. Binary
3. Two-Hot (new technique)

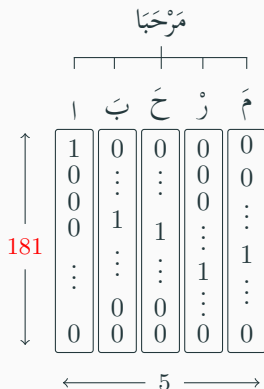
# One-Hot



181 is the number of all combination between letters and diacritics.

$$181 = 36 + 36 \times 4 + 1$$

# One-Hot, example

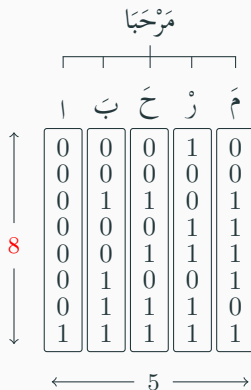


For English, vector length is 28.

# Binary

Let  $n$  be the vector length.

$n = \lceil \log_2 l \rceil$   $l \in \{181, 28\}$  for Arabic and English alphabet, respectively.



# Pros and Cons

## One-Hot:

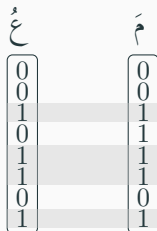
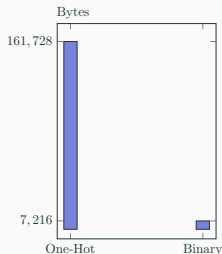
- $181 \times 1$  too long vector.
- Wast of memory, compared towith Binary.

But it's very informative and easy to learn.

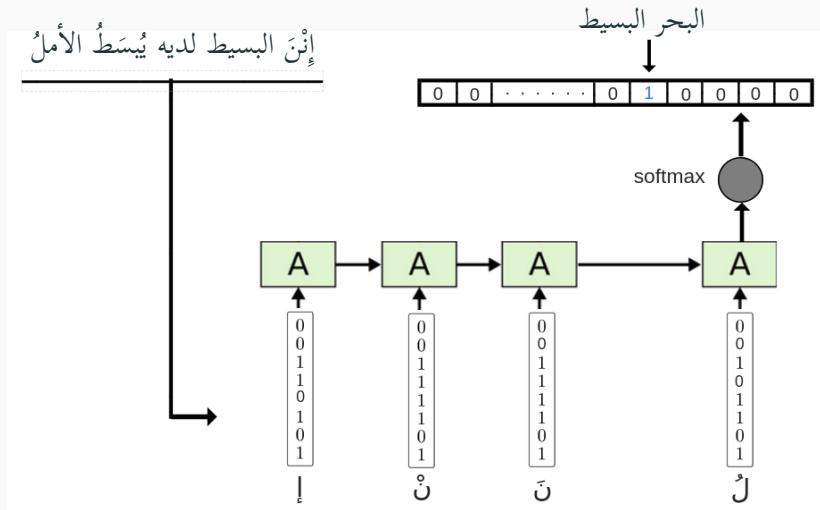
## Binary:

- The problem of a common feature, the optimizer updates weights of this two common features to reduce the error of one character of them it will affect the rest characters that have the same feature.

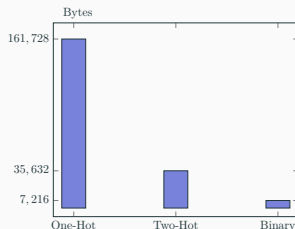
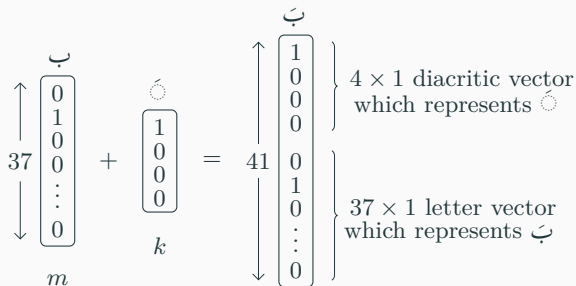
But, memory is well consumed.



# Feeding verse to model



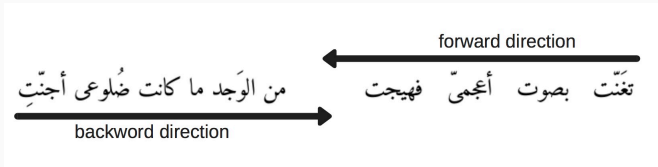
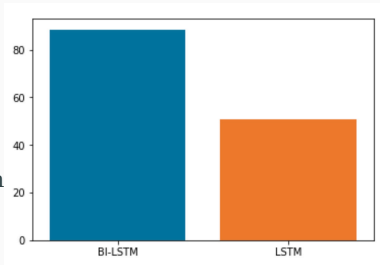
# Two-Hot Encoding



- Memory is well consumed.
- Have only one common feature.
- For Arabic only.

# Arabi Results

- Bi-LSTM models always outperform LSTM models.
- It means that models can't learn the pattern from one direction, it should be two directions together.





# Arabi Results

- Model 1 is the best model **with** diacritics.
- Model 2 is the best model **without** diacritics.
- Both models are **Two-Hot**.
- Low accuracy for the last 4 meters, because the data-points of their classes are very small.

Class	Model 1	Model 2
Wafeer	95.70%	95.89%
Monsareh	89.34%	89.48%
Madeed	83.79%	81.28%
Mogtath	85.10%	84.53%
Motakarib	95.91%	96 %
Kamel	96.49%	96.74%
Taweel	98.01%	97.81%
Sar'e	91.86%	90.18%
Raml	93.45%	92.98%
Rigz	89.46%	86.12%
Khafeef	96.67%	96.59%
Baseet	98.05%	98.03%
Moktadib	71.43%	68.37%
Hazg	84.81%	77.79%
Modar'e	33.33%	20.83%
Motadarik	83.43%	78.88%

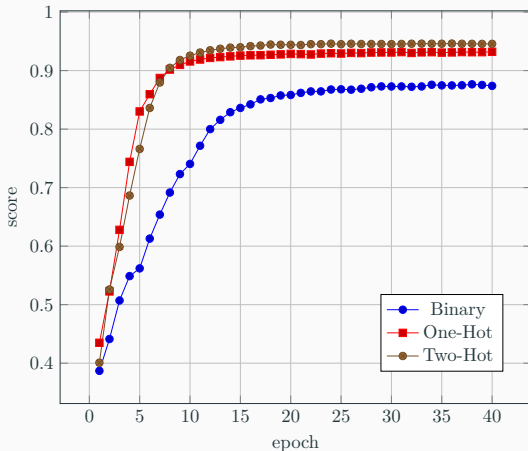
# Encoding Effect

- In fact, the three encoding achieve the same results but with different architecture in case of data with **diacritics**.
- But in case of data without **diacritics** the Two-Hot is the best.
- Binary encoding needs **deeper** architecture to achieve high results.

	diacritics	
encoding	yes	no
Two-Hot	95.79%	95.43%
8bit	95.51%	93.21%
One-Hot	95.32%	93.94%

# Encoding Effect

- When applying the same architecture but changing one factor only, **Encoding**.
- **Two-Hot** is the best.
- **One-Hot** is very close to **Two-Hot**.



# Diacritics effect

- On average, **two-hot** models achieve the same results regardless the **diacritic** state.
- **Binary** and **One-Hot** achieve higher results in case of diacritics data than without diacritics.
- Diacritics increases **Binary** and **One-Hot** learning accuracy. But it does not matter in case of **Two-Hot**.

encoding	diacritics	
	yes	no
Two-Hot	95.79%	95.43%
8bit	95.51%	93.21%
One-Hot	95.32%	93.94%

# Diacritics effect

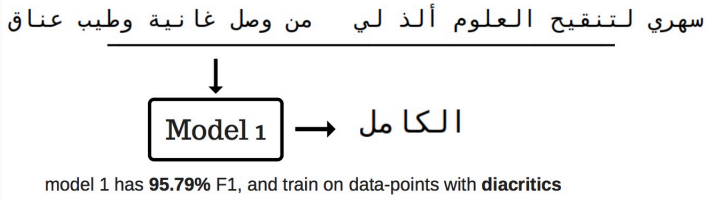
سَهْرِي لِتَنْقِيحِ الْعُلُومِ أَلَدُّ لِي مِنْ وَصَلِ غَايَةِ وَطِيبِ عِنَاقِ



Model 1 → الكامل

model 1 has **95.79%** F1, and train on data-points with **diacritics**

- Model 1 assigned الكامل to this verse by 99.26%.
- Model 1 achieves F1=91.62% with the same test set but without diacritics.



- Model 1 assigned الكامل to this verse by 98.18%.
- Model 1 achieve F1=91.62% with the same test set but without diacritics.

id	encoding	cell type	f1 test
1	one-hot	GRU	81.35%
2	one-hot	LSTM	80.34%
3	binary	LSTM	75.43%
4	binary	GRU	75.04%

Questions?