إعطاء عنوان لنص معين

امل حجازي، براء الشيخة، سيلفي كبه، شربل قلومة

قسم الذكاء الصنعي، جامعة دمشق

ملخص العمل :

الهدف من المشروع هو توليد عنوان مناسب لنص معين حيث قمنا بتجهيز وتقسيم البيانات ومن ثم تدريب على عدة نماذج وبعدها اخذ خبر معين من test وايجاد topic الاعلى للنص المراد وضع عنوان مناسب له و ايجاد الكلمات الاكثر اهمية من هذه المواضيع ثم ادخالها الى النماذج السابقة لتشكيل العنوان وثم قياس مدى تقارب هذه العنواين مع النص المعطى وتم اعطاء عناوين باستخدام aragpt2 model

الكلمات المفتاحية : ***nlp generate title paper***

I. المقدمة:

ان تسمية مشروعك الجديد قبل طرحه للاسواق او تسمية منتج او عنونة ورقة البحث تكون عادة من اصعب مراحل العمل لانه عليك اختيار اسماء لافتة للقارئ او للزبون و من هنا جاءت فكرة اعطاء عنوان لنص معين حيث يعد مبدأ توليد عنوان لنص احدى وظائف معالجة اللغات الطبيعية و لانشاء model يولد العناوين علينا ان ندرب المودل على معرفة احتمالية الكلمة من خلال استخدام الكلمات التي ظهرت في تسلسل النص كسياق .
يوجد العديد من البرامج او التطبيقات او الادوات المساعدة لتوليد اسم او عنوان من عدة كلمات او نص يدخله المستخدم و هي تعطيه عدة نتائج محتملة.

II. الأعمال السابقة:

- <u>المشاريع السابقة:</u>

A. Project name generator[1] :

يستخدم هذا التطبيق لإنشاء أسماء مشاريع لجميع مشاريعك ، سواء كانت مشاريع عمل أو مشاريع جانبية.
يقدم التطبيق ثلاث طرق لتوليد أسماء المشاريع.
- أولاً عن طريق تحديد حرف البداية أو أحرف البداية واختيار الكلمات التي تم إنشاؤها عشوائيًا بدءًا من هذه الأحرف.
- ثانيًا عن طريق أداة التوزيع العشوائي التي تسمح لك بإدخال كلمة وستقوم بترتيب كلمتك عشوائيًا لإنشاء اسم المشروع.
- ثالثا عن طريق تحديد مجموعة او مجموعات لاختيار كلماتك منهم لانشاء الاسم الجديد

B. Namify[2]:

هو ايضا اداة مولدة للاسماء مجانية عبر الانترنت تتيح لك الوصول إلى آلاف الأفكار لاسم المشروع وفقًا للمعلومات التي تدخلها في شريط البحث الخاص بها، بمجرد إدخال الكلمات الرئيسية ذات الصلة واختيار الصناعة ، ستتمكن من تصفح قائمة طويلة وشاملة من الأسماء القيمة، و هنا بعض مميزاتها:
-الاسماء ذات الصلة مضمونة :تقوم تقنيتها بفرز الاسماء ذات الصلة و غير الصلة و تقدم لك فقط الاسماء ذات المغزى و المعقولة في النهاية.
-الاسماء الفريدة: يمكنك توقع اسماء مميزة لاتنسى اي صعب نسيانها لدى الزبون اوالمتلقي.
-Domain names: لا تقلق بشأن توفر أسماء النطاقات لاسمك الذي اخترته ، تعرض لك Namify امتدادات المجالات المتاحة لكل اسم (مثل .store و .online و .tech و .site و .fun و .space و .uno وما إلى ذلك ، حتى لا تضطر إلى التنازل عن اختيارك اسم.)
-و من اهم مميزاتها Namify تقدم لك شعارًا مجانيًا لتعزيز قيمة مشروعك بشكل أكبر.

C. Namelix[3]:

بالنسبة للشركات الجديدة ، قد تبدو خيارات التسمية محدودة للغاية لان النطاقات القصيرة باهظة الثمن و في نفس الوقت الأسماء الطويلة متعددة الكلمات لا توحي بالثقة لذلك Namelix :
- يولد أسماء قصيرة وجذابة بحيث كلما كانت كلماتك الرئيسية أكثر تحديدًا ، كانت نتائج الاسماء أفضل.
- تجمع معظم أدوات إنشاء الأسماء التجارية بين كلمات القاموس لإنشاء أسماء أطول،بينما ينشئ Namelix أسماءًا قصيرة ذات علامات تجارية واضحة لها صلة بفكرة عملك.
- يمكنك ان تقرر ما إذا كنت تعطي الأولوية لاسم أقصر ، مع وجود كلمة رئيسية معينة أو امتداد المجال
- تتعلم الخوارزمية الخاصة به من الأسماء التي تعجبك ، مما يمنحك توصيات أفضل بمرور الوقت

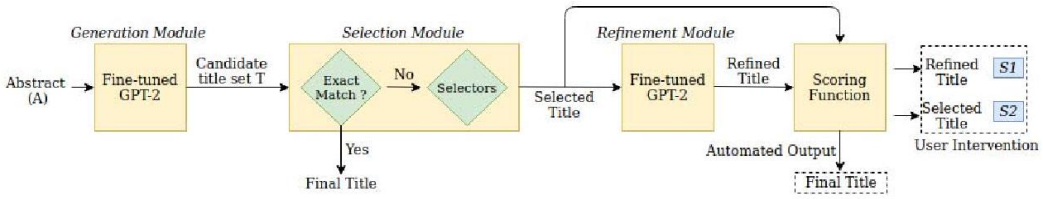
D. Namebot[4]:

هي أداة تقوم بإنشاء مئات الآلاف من الأسماء الجذابة و القصيرة و ذات معنى باستخدام كلمة أو أكثر من الكلمات مولدة الدلالية، يمكنها العثور على أسماء مثالية لأي نوع من الأعمال أو تطبيقات الهاتف وهي مجانية تمامًا ،مميزاتها :
-تولد البدائل الأكثر شيوعًا والمدهشة لاسم المجال الخاص بك باستخدام المرادفات والمتضادات من قاعدة بيانات الكلمات التي تحتوي على الملايين
- الجمع بين الكلمات ودمجها لإنشاء أسماء ذات توجه عالٍ للعلامة التجارية ، وأيضًا استخدام الأخطاء الإملائية وعناوين URL القصيرة والتداخل والاختصارات والتنوعات الصوتية.
- تستخدم بادئات ولاحقات القاموس الشائعة مع الكلمة الأساسية ككلمة جذر لإنشاء اسم مجال مثالي موجه لتحسين محركات البحث.

- Paper التي تم القراءة عنها:

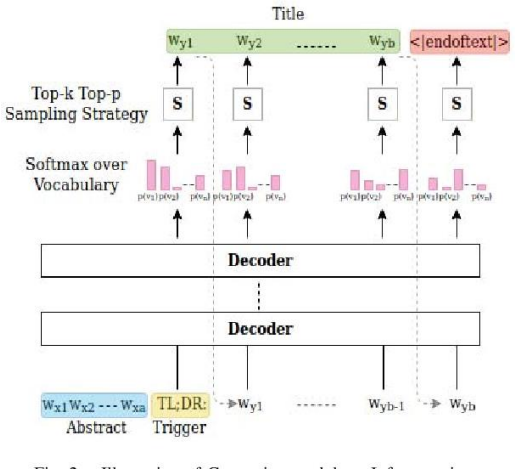
E. Generate automatic title for text Transformer language model [5] :

❖ يتكون pipeline الخاص ب model من ثلاث وحدات :



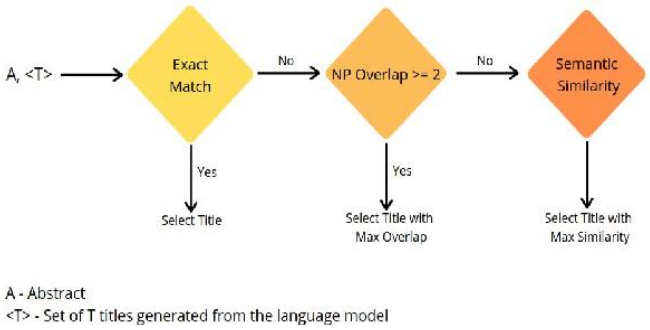
1. Generation Module:

يحوي على شبكة GPT-3 والتي تقوم بتوليد عنواين بشكل تلقائي لنص معين الخرج الخاص به في مجموعة من من العنواين T يتم بعدها اعطاء احتمال لكل جملة من خلال SoftMax وبعدها اختيار اعلى كلمات ذات الاعلى احتمال من الجمل باستخدام Top_k Top_p ومن ثم توليد عنوان واحد ولمعرفة وجود تداخل بين الكلمات يتم تمرير العنوان الى قسم selection



2. Selection Module:

في هذا القسم يتم معرفة في حال وجود تداخل في الكلام يتم تمرير العنوان T الذي تم توليده في القسم السابق يتم ذلك من خلال عمل match بين العنوان والنص الموجود في حال وجودها يختار العنوان المعين اما في حال قيمة عدم التشابه اكبر من 2 يختار العنوان الذي لديه Max overlap عالي اما في حال اقل من 2 نختار العنوان الذي يحوي على Max Similarity بالمعنى الدلالي والهدف من هذا القسم التاكد من ان العنوان المنشأ لا يحوي اخطاء بالمعني الصرفي والدلالي



3. Refinement Module:

تحوي على شبكة gpt-3 ايضا الهدف منها هو بعد اختيار عنوان واحد من الخطوة السابقة في حال وجود نقص بمعنى الجملة او وجود كلمات ناقصة مثال t1 (p) t2 بالموضع p لا يوجد كلمة فيقوم بتوليد هذه الكلمة وايضا في حال وجود كلمات زائدة بالجملة يقوم بحذف هذه الكلمات وبينتج منها Refined Title

4. Scoring Function:

في هذه الخطوة يتم اعطاء النص بشكل تلقائي من دون تدخل الانسان لكن في حال وجود تدخل بشري نقوم بعرض العنوان T الناتج عن الخطوة الاولى والعنوان F الناتج عن الخطوة الثالثة مع احتمال كل منها Rs وبعدها نقوم باختيار احد العنوانين من قبل الانسان

III. الطريقة **Methodology**:

- جمع وتقسيم البيانات (data set):

البيانات بمشروعنا عبارة عن مجموعة من الاخبار العربية تم تجميعها من مقالات مع اخذ العناوين الخاصة بكل مقال ومن ثم تخزينها في data set بلغ عدد الاخبار حوالي 16000 تم تقيسمها الى train, test بنسبة (0.6,0.4) وتحوي على عامودين العنوان الخاص بالخبر اضافة الى الخبر نفسه (story, title) تم التدريب على title و story ومقارنة النتائج بينهم

- تنضيف البيانات (clean the train data set):

تم اختيار خطوات تنضيف الداتا وذلك من خلال تثبيت نموذج base line ومراقبة النتائج لاختيار الخطوات الافضل

1) عمل tokenization للعنوان من خلال استخدام تابع arabertv2 model
2) ازالة الاحرف الغير عربية واضافة الى الارقام
3) ازالة علامات الترقيم والمحارف الغير ضرورية (punctuation) وازالة الفراغ من الجمل
4) ازالة العنواين والصور والهاشتاغ
5) توحيد بعض الاحرف وازالة الاحرف الممدودة وحذف الاحرف المتكررة وذلك لتوحيد الكتابة قدر المستطاع
6) استخدام ISRIStemmer لايجاد جذر الكلمات باللغة العربية

- تجهيز train data set لتدريبها:

1) تحويل الداتا بعد التنضيف الى tokens
2) تشكيل n_grame بطول العنوان وذلك لتشكل input sequences
3) توليد padding للعنواين بطول اطول عنوان وذلك لتوحيد الطول للعنواين
4) تقسم train data set الى x_train و y_train

- تجهيز test data set:

لتوليد العنوان نحتاج لايجاد الكلمات المهمة في نص المعطى لايجاد عنوان مناسب له قمنا بتحقيق هذه الخطوة بايجاد الكلمات الاكثر اهمية في النص باستخدام BERTopic بعد محاولة عدد من الطرق من ضمنها LDA وملاحظة انه يعطي نتائج افضل بعدها يتم تشكيل n-grams للكلمات المهمة وتمريرها على النماذج المستخدمة لاستخراج العنوان المناسب .

- تم تنفيذ الخطوات السابقة على story بدلا من title.

IV. التجارب:

تم اختيار قيم المعاملات hyperparameters باستخدام gird search حيث تم وضع عدة قيم تبعا للنموذج المستخدم مع تعديل على هذه القيم لحل مشكلة overfitting

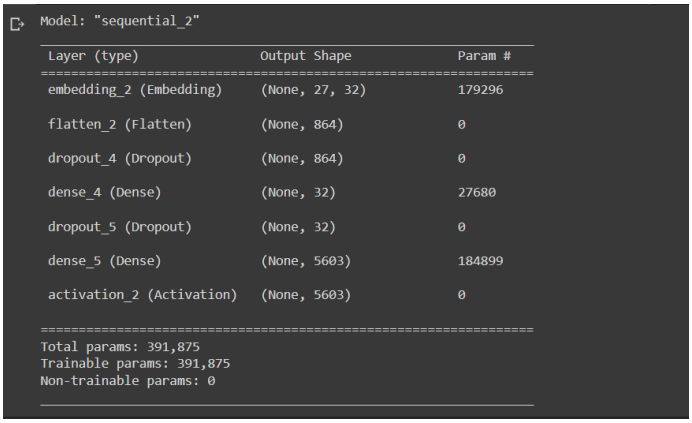
- Model for title:

1. Base line:

- شرح عن طبقات النموذج:

تم استخدام نموذج أولي base line وذلك لمقارنة نتائجه من نتائج النماذج الاخرى يتالف من طبقة embedding وطبقة flatten وطبقة hidden واحدة وتم وضع طبقتين Dropout الطبقة الاولى بقيمة 0.5 اما الطبقة الثانية بقيمة 0.2 لحل مشكلة overfitting وايضا استخدام Model Checkpoint لحفظ الاوزان الاقل خطا واستخدام تابع الخطا (categorical_crossentropy)

: Summary for model



```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_2 (Embedding)     (None, 27, 32)            179296

 flatten_2 (Flatten)         (None, 864)               0

 dropout_4 (Dropout)         (None, 864)               0

 dense_4 (Dense)             (None, 32)                27680

 dropout_5 (Dropout)         (None, 32)                0

 dense_5 (Dense)             (None, 5603)              184899

 activation_2 (Activation)   (None, 5603)              0

=================================================================
Total params: 391,875
Trainable params: 391,875
Non-trainable params: 0
_________________________________________________________________
```

- قيم المعاملات النهائية hyperparameters:

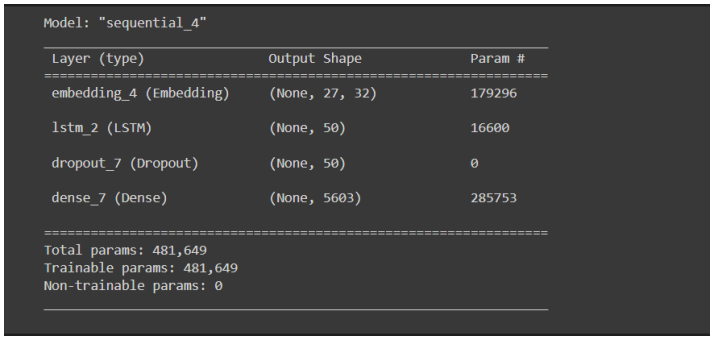
{ embedding_dim: 32, hidden1: 32 }

.2 model with LSTM:

- شرح عن طبقات النموذج:

تم اضافة طبقة lstm عوضا عن طبقة ال hidden اضافة الى طبقات النموذج السابق

: Summary for model



```
Model: "sequential_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_4 (Embedding)     (None, 27, 32)            179296

 lstm_2 (LSTM)               (None, 50)                16600

 dropout_7 (Dropout)         (None, 50)                0

 dense_7 (Dense)             (None, 5603)              285753

=================================================================
Total params: 481,649
Trainable params: 481,649
Non-trainable params: 0
_________________________________________________________________
```

- قيم المعاملات النهائية hyperparameters:

{ embedding_dim: 32 , 'num_Lstm': 50}

.3 model with Bidirectional LSTM and LSTM:

- شرح عن طبقات النموذج:

تم اضافة Bidirectional LSTM على النموذج السابق لتصبح ال summary الخاصة به :

```
Model: "sequential_10"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_10 (Embedding)    (None, 27, 32)            179296

 bidirectional_6 (Bidirectio  (None, 27, 64)           16640
 nal)

 dropout_16 (Dropout)        (None, 27, 64)            0

 lstm_14 (LSTM)              (None, 64)                33024

 dense_13 (Dense)            (None, 5603)              364195

=================================================================
Total params: 593,155
Trainable params: 593,155
Non-trainable params: 0
_________________________________________________________________
None
```

- قيم المعاملات النهائية hyperparameters:

{num_Lstm2': 64, 'num_Lstm1': 64, 'embedding_dim': 50'}

.4 model with GRU:

- شرح عن طبقات النموذج:

تم اضافة طبقة GRU عوضا عن lstm لتصبح ال Summary الخاصة به :

```
Model: "sequential_11"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_11 (Embedding)    (None, 27, 32)            179296

 gru (GRU)                   (None, 32)                6336

 dropout_17 (Dropout)        (None, 32)                0

 dense_14 (Dense)            (None, 5603)              184899

=================================================================
Total params: 370,531
Trainable params: 370,531
Non-trainable params: 0
_________________________________________________________________
None
```

- قيم المعاملات النهائية hyperparameters:

{ embedding_dim: 32 ,GRU: 32}

.5 model with Attention and Lstm:

- شرح عن طبقات النموذج:

تم اضافة طبقة attention الى نموذج Bidirectional LSTM and LSTM لتصبح ال Summary الخاصة به:

```
Model: "sequential_15"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_15 (Embedding)    (None, 27, 32)            179296

 bidirectional_10 (Bidirecti  (None, 27, 64)           16640
 onal)

 attention_3 (Attention)     (None, 27, 64)            91

 lstm_22 (LSTM)              (None, 32)                12416

 dense_18 (Dense)            (None, 5603)              184899

=================================================================
Total params: 393,342
Trainable params: 393,342
Non-trainable params: 0
_________________________________________________________________
```

- قيم المعاملات النهائية hyperparameters:

{ embedding_dim: 27 , num_Lstm1: 64, 'num_Lstm2 :32}

.6 aragpt2 model :

هو نموذج مشابه gpt-2 model لتوليد نصوص باللغة العربية وهو عبارة عن نموذج مدرب مسبقا على مجموعة من البيانات باللغة العربية ، قمنا باستخدامه بمشروعنا للحصول على نتائج افضل

V. النتائج:

- آلية تقييم العمل:

تم حساب الدقة accuracy لكل نموذج لكنها لا تعتبر معيار لتقييم العمل حيث التقييم المناسب لمشروعنا هو الاعتماد على مدى قدرة النموذج على تشكيل عناوين مناسبة

- جداول مقارنة النماذج المختلفة :

.1 Model for title:

	model_name	preprocessing_methods	validate accuracy
0	model with flatten and one hidden layer	with some proccess function	0.3050
1	model with Bidirectional lstm	with some proccess function	0.2706
2	model with lstm	with some proccess function	0.3188
3	model with GRU	with some proccess function	0.2623
4	model with Attention and lstm	with some proccess function	0.2955

2. Model for Story:

تم اعتماد على معيار ppl للمقارنة بين النماذج وهو ليس كافي لقياس مدى قدرة النموذج على توليد عنوان مناسب

	model_name	preprocessing_methods	validate ppl
0	model with lstm	with some proccess function	1.0622
1	model with flatten and one hidden layer	with some proccess function	1.0636

- مقارنة نتيجة العمل:

نتائج توليد نص باستخدام كل نموذج :

على الرغم من مشاهدة نتائج جيدة نوعا ما باستخدام data set باللغة الانكليزية لم يتم توليد نتائج جيدة باللغة العربية من الاقتراحات التي قمنا بمحاولتها لتحسين النتائج:

- زيادة حجم epoch لكن ادى ذلك لحدوث overfitting
- زيادة حجم train data وذلك لزيادة البيانات للتعلم لكن لم يتحسن الوضع
- محاولة تنظيف الداتا باكثر من طريقة من خلال اضافة وحذف التوابع ومراقبة النتائج وحصلنا على افضل النتائج باستخدام التوابع السابقة , النتائج التي تم الحصول عليها:

1. Base line :

```
الي ''
متسائل ''
حسب ''
مهني ''
طرق ''
راح ''
```

شرح النتيجة:

السبب لعدم وجود طبقة تلافيفية وبالتالي لم يستطيع ايجاد الترابط بين الكلمات حسب السياق لذلك اعطى جواب واحد وهو فراغ لكل الكلمات

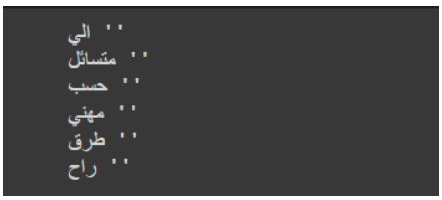
2. model with LSTM:

```
'الي ال
'متسائل ال
'حسب ال
'مهني ال
'طرق ال
'راح ال
```

شرح النتيجة:

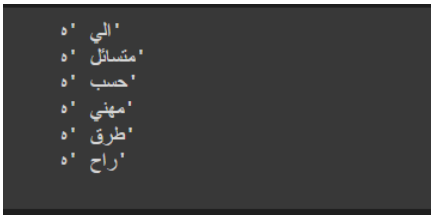
على الرغم من استخدام طبقة lstm الا ان النتائج لم تكن جيدة حيث لم يقم بتوليد كلمة على الرغم من عدم وجود overfitting

3. model with Bidirectional LSTM and LSTM:



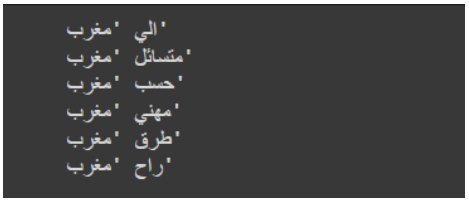
شرح النتيجة:
لم يقم بتوليد نتائج افضل بل قام بتوليد فراغ مثل نموذج base

4. model with GRU:



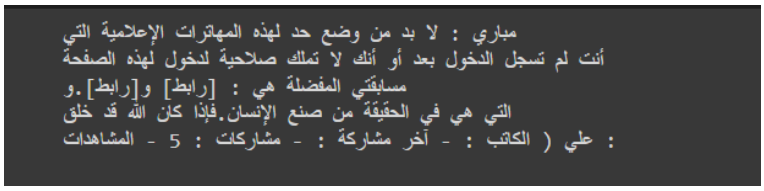
شرح النتيجة:
يشبه نتائج نموذج lstm لم يقم بتوليد الا حرف واحد

5. model with Attention and Lstm:

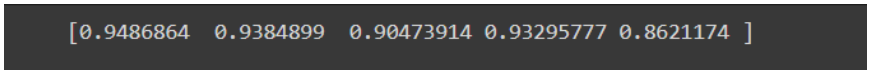


شرح النتيجة:
افضل النماذج التي تم تدريبها سابقا حيث قام بتوليد كلمة و ليس حرف او فراغ ،لكن هناك مشكلة لم يقم بتوليد الا كلمة واحدة

6. aragpt2 model:



شرح النتيجة:
من افضل النماذج التي اعطت نتائج جيدة حيث قام بتوليد نص كامل لكن يحتاج الى معالجة للحصول على نتيجة افضل , هنا قمنا بايجاد الترابط بينه وبين النص الاصلي وذلك باستخدام bert model لايجاد embedding للكلمات وبعدها استخدام cosine_similarity لايجاد الترابط بينهم وكانت النتيجة ليست منطقية كثيرا :



احد الحلول استخدام model اخرى عوضا عن bert لايجاد embedding

IV. الخلاصة:

تطبيق عدد من النماذج لم نقم بتطبيقها من قبل واضافة الى حل العديد من المشكلات للحصول الى الحل الافضل كمشكلة overfitting و ايضا امكانية اختيار المعاملات hyperparameters المناسبة للداتا بناءا على المشكلة المقترحة واختيار المعاملات الافضل للنموذج المستحدم

- الآفاق المستقبلية :

توليد عنواين مميزة للشركات لجذب الزبائن وذلك من خلال امكانية فرز العنواين ذات الصلة وتصحيح اخطاء العنوان المولد

- ما لم يسعنا الوقت لتحقيقه لضيق الوقت:
  - حساب مقدار التشابه بين الجمل حسب السياق واختيار الجملة الاكثر تقاربا
  - معالجة الجملة المختارة من خلال معرفة اذا كانت الكلمات المولدة مناسبة لغويا ام لا
  - ازالة الكلمات الغير ملائمة ومن ثم توليد الكلمات المفقودة
  - قياس مدى تقارب الجملة الناتجة عن الجملة الاصلية وضع score معين للتقارب
  - اخذ التقارب الافضل الاعلى دقة في حال عدم وجود تدخل بشري

المراجع:

[1] https://apkpure.com/project-name-generator/com.csrhymes.projectnamegenerator

[2] https://namify.tech/project-name-generator

[3] https://namelix.com

[4] https://www.namobot.com

[5] https://ieeexplore.ieee.org/document/9364613