مجلة جامعة طرطوس للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (6) العدد(3) 2022
Tartous University Journal for Research and Scientific Studies - engineering Sciences Series Vol. (6) No. (3) 2022

طرائق جديدة لطرق تصنيف النصوص

د. جعفر سلمان *

(تاريخ الإيداع 14 /2/ 2022 . قُبِل للنشر في 18 /4/ 2022)

□ ملخّص□

إن أحد أكثر أشكال تنظيم البيانات المخزنة شيوعًا في هذه الأيام هو الشكل النصي ، وتعد القوانين والقواعد والإحصاءات المختلفة للكوارث الطبيعية والمعلومات من صنع الإنسان وغيرها من المؤشرات أمثلة على المعلومات النصية. تراكمت كمية هائلة من المعلومات النصية لعقود عديدة في مختلف الصناعات ومجالات الإنتاج والتي تحتاج إلى المعالجة، وعلى وجه الخصوص تصنيف المستندات المختلفة وإنشاء تسلسلات هرمية لها حسب الموضوع.

تم اقتراح زيادة الملخص موضحا ما تم إنجازه في هذا البحث بشكل أوسع.

الكلمات المفتاحية: معلومات نصية، تخزين معلومات، تصنيف، طريقة تعلم الآلة.

* مدرس في قسم هندسة تكنولوجيا المعلومات – كليّة هندسة تكنولوجيا المعلومات والاتصالات – جامعة طرطوس– سوريا

مجلة جامعة طرطوس للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (6) العدد(3) 2022
Tartous University Journal for Research and Scientific Studies - engineering Sciences Series Vol. (6) No. (3) 2022

New Approaches to the task of Text Classification

Dr.Jaafar Salman *

(Received 14/ 2/ 2022 . Accepted 18/ 4/ 2022)

□ ABSTRACT □

Today, one of the most common forms of data storage organization is text form. Laws, regulations, various statistics of natural and man-made disasters and other indicators are all examples of text information. For many decades, a huge amount of text information has been accumulated in various industries and areas of production, which needs to be processed, in particular, to classify various documents and create hierarchies of them by subject. This work is dedicated to this task.

Key Words: text information, information storage, classification, machine learning Alogorithms

*Teacher, Information Technology Engineering Department, Information and communication Technology Engineering , Tartous University, Syria .

1- مقدمــة

يعد حجم البيانات والمعلومات النصية التي يتعين على المنظمات والشركات العمل معها هائل جداً ويمكننا أن نذكر على سبيل المثال: قاعدة بيانات الوثائق القانونية لوزارة العدل ووزارة الزراعة والبيئة ووزارة العمل والشؤون الاجتماعية (والتي تحوي ملايين المستندات للموظفين والعاملين في الدولة والقطاع الخاص)، بالإضافة إلى العديد من المنتجات المنشورة من قبل الوزارات والإدارات ذات الصلة في مختلف المجالات. لذلك فإن حجم البيانات ضخم ويجب التكيف مع معالجتها وتحديثها (على سبيل المثال إضافة نسخ رقمية من المقالات العلمية القديمة إلى قاعدة البيانات مع وضع علامات على المجالات العلمية التي تنتمي إليها). يمكن أن نقول هنا أن هذا المجال هو المكان الذي سيكون فيه تطبيق أساليب التعلم الآلي machine learning مفيد جداً.

2- هدف البحـث

يهدف البحث إلى النظر في المهام المطبقة لتصنيف النصوص غير المنظمة من خلال طرق الأتمتة automation باستخدام أساليب تعلم الآلي والنظر في الأساليب الأساسية في مجال الحلول المقترحة للمشكلة المدروسة والتحقق من خلال الاختبار والتقييم التجريبي لجودة الأساليب المدروسة

3- طرائـق البحـث ومـواده

أُنجز هذا البحث اعتماداً على دراسات ومراجع علمية حديثة وعديدة تختص في هذا المجال وقد أخذت نتائجها و توصياتها بعين الاعتبار[4, 5, 7,10] .

3-1- الخوارزمية المقترحة Suggesting algorism

بناءً على تحليل المقالات المتعلقة بالموضوع قيد البحث فقد تم تحديد الخطوات التالية في مهمة تصنيف النص:

1. إعداد مجموعة من النصوص التدريبية training set of texts: من الضروري جمع مجموعات النصوص وتحديد الموضوعات التي تنتمي إليها هذه النصوص يدويًا.
2. اقتطاع من النصوص ما يسمى بكلمات التوقف stop-words التي تستخدم بشكل متكرر والتي تظهر في جميع النصوص ولا تحتوي على أي معلومات حول الموضوع الذي ينتمي إليه النص (على سبيل المثال الكلمات الرابطة connective words).
3. نقسّم كل نص إلى كلمات منفصلة separate words ونقوم بتقييسها normalize (على سبيل المثال، إزالة البادئات غير الضرورية وترجمتها إلى حالة اسمية، وما إلى ذلك).
4. تحويل النص الذي تم تسويته إلى متجه عددي numerical vector.
5. تدريب المصنف train the classifier.
6. تنفيذ الخطوات من 1–4 لتصنيف نص جديد
7. استخراج الصيغة الخوارزماتية في كتابة الخوارزمية (اسم المصدر في بداية الجملة)

سنوضح الخطوتين 4 و5 بمزيد من التفصيل: تحويل النص إلى متجه سمات مجموعة من الكلمات: تحوي كل كلمة من مجموعة التدريب لعملية المقارنة بعض المعرفات غير السلبية non-negative identifier i، يتم حساب عدد التكرارات n لكل كلمة ضمن المستند j، بعد ذلك نشكل المصفوفة M(i, j) = n لنحصل على مصفوفة متفرقة

sparse matrix (تحتوي على العديد من الأصفار) وبعداً ضخماً huge dimension (عادةً ما يصل عدد الكلمات الفريدة إلى عدة عشرات الآلاف).

حصلنا ضمن الخطوة السابقة على عدد تكرارات الكلمات ضمن النص، لكن في النصوص الأطول سيكون عدد تكرارات الكلمات أكبر لذلك سيكون من المستحيل مقارنة النصوص ذات الأطوال المختلفة دون التبديل من عدد التكرارات number of occurrences إلى تردد التكرارات frequency of occurrences (لتوضيح الحاجة إلى التبديل إلى التردد نأخذ النص X ونشكل منه النص Y وهو تسلسل concatenation النص X مع نفسه. نحدد متجه الميزات V(X) = ½ V(Y) مما يعني أن V (X) و V (Y) يمكن أن يكونا مختلفين تماماً في القاعدة، على الرغم من أنه من الواضح أن النصوص يجب أن تنتمي إلى نفس الموضوع)[1, 2, 3] .

تجدر الإشارة إلى أنه في مشاكل تصنيف المعلومات غير النصية non-text information غالبًا ما يتم استخدام النهج التالي لتقليل مساحة الميزات space of features (تقليل أبعاد مساحة الميزة): يتم تثبيت الميزات الشائعة جداً في التصنيف فقط (في حالتنا يجب تحديد الكلمات الأكثر شيوعًا).

1-1-3 تموضع القيمة المفرد (SVD) + singular (LSA) Latent semantic analysis value decomposition (SVD)

نقصد بالـ SVD النسخة المصغرة reduced version حيث لن نقوم بمعالجة جميع القيم المفردة ولكن فقط k من أعلى القيم، k هو بارامتر يتم تحديده تجريبياً. يمكن استخدام هذه الطريقة كطريقة لتحويل مجال السمة الأصلي الذي تم الحصول عليه بحقيبة من الكلمات a bag of words إلى مجال سمة دلالية semantic attribute بأبعاد أقل بشكل ملحوظ. بالإضافة إلى تقليل أبعاد مساحة الميزة فإن هذا التحول يعالج بنجاح آثار المرادفات والتكافؤ المتعددsynonymy and multivalence وكلاهما يحدد معاني مختلفة لكل كلمة مما يؤدي إلى أن تصبح مصفوفات المصطلحات متباعدة بشكل كبير ولديها تشابه ضعيف poor similarity إلى حد ما وفقاً لمقاييس مثل مقياس تشابه التجيب cosine similarity measure [4, 5].

من وجهة النظر الرياضية، فإن SVD المختصر المطبق على مجموعة التدريب يؤدي إلى تقريب المصفوفة من X

$$X \approx X_k = U_k \ \Sigma_k \ V_k \qquad (1)$$

حيث تكون المصفوفة Σ قطرية، $\Sigma_{ii} = \sqrt{\lambda_i}$, λ_i هي القيم الذاتية للمصفوفة $X_k X_k^T = X_k^T X_k$ [6].

يتم استخدام $U_k \sum_k^T$ كمتجه للميزات مع k ميزة. لتحويل المصفوفة الجديدة للمصطلحات Y التي تم الحصول عليها من مجموعة التدريب، نحتاج فقط إلى ضربه فيV_k على اليمين. في الواقع فإن SVD المختصر يشبه إلى حد بعيد طريقة PCA التي تنتشر على نطاق واسع في مشاكل تقليل بُعد فضاء الميزة، وعلى عكس ذلك فهي تعمل مباشرة على مصفوفة المصطلحات وليس مصفوفة التغايرات.

2-3- التصنيف Classification

1-2-3- تصنيفات SVM+SGD

يعرف انحدار التدرج العشوائي Stochastic Gradient Descent (SGD) بأنه نهج بسيط ولكنه فعال للغاية لتدريب المصنفات الخطية المميزة بتوابع أفضلية محدبة convex penalty functions مثل طريقة المتجه المرجعي reference vector method والانحدار المنطقي logistic regression. على الرغم من أن SGD قد تم تطويره منذ وقت طويل إلا أن اهتمامه به زاد مؤخراً في سياق التدريب الواسع النطاق.

تم استخدام SGD بنجاح لمهام التعلم الآلي عالية الأبعاد والمتفرقة وغالبًا ما يستخدم في تصنيف النصوص ومهام معالجة اللغة language processing tasks. يمكن للمصنفات من هذا النوع التعامل بسهولة مع المهام التي تحتوي على أكثر من 5 ^ 10 أمثلة تعليمية وأبعاد فضاء الميزة أكثر من 5 ^ 10 [7] .

فوائد انحدار التدرج العشوائي هي:

- الكفاءة efficiency (بفرض مصفوفة التعلم بالحجم (n, p) وعدد التكرارات k عندها يكون التعقيد مساوياً $O(knp\bar{)}$، حيث $\bar{p}$ هو متوسط عدد العناصر غير الصفرية لكل مستند (عمود ضمن المصفوفة)).
- سهولة التنفيذ ease of implementation.

يمكن أن تُعزى عيوب disadvantages انحدار التدرج العشوائي إلى:

- يجب تحديد العديد من البارامترات مثل بارامتر التنظيم regularization parameter وعدد التكرارات number of iterations.
- يعد SGD حساس للقياس sensitive to scaling (من الضروري تحديد كيفية تسوية متجهات الميزات).

من وجهة النظر الرياضية، فإن مشكلة المصنفات الخطية لها الشكل التالي: سنقوم بالتوضيح من خلال مجموعة من الأمثلة التدريبية $(x_1, y_1), \dots, (x_n, y_n)$، حيث $x_i \in R^n$ $y_i \in \{-1,1\}$ (تصنيف ثنائي binary classification) [8, 9, 10]، نبحث بعد ذلك عن دالة حاسمة decisive function من شكل $f(x) = w^T x + b$ مع متجه من البارامترات $w \in R^m$ (تم الحصول عليها في مرحلة التدريب النموذجي) وبعض الإزاحة $b \in R$. من أجل اتخاذ القرار سننظر إلى إشارة الدالة عند استبدال متجه جديد والطريقة الأكثر شيوعًا للعثور على البارامترات هي تقليل خطأ التدريب المنتظم [11، 12]:

$$E(w,b) = \frac{1}{n}\sum_{i=1}^{n} L\big(y_i, f(x_i)\big) + \alpha R(w) \qquad (2)$$

حيث L هي دالة خسارة، تقيس درجة "الجودة المنخفضة poor quality " للنموذج وR هي دالة أفضلية penalty function تصف تعقيد العملية (كلما كانت قيمتها أعلى تصبح العملية أكثر استقراراً وتقل الإصدارات emissions غير الطبيعية)، $\alpha > 0$ هي فقط بعض البارامترات غير السلبية.

يعتمد نوع المصنف الذي سنحصل عليه عند الإخراج على اختيار التابع:

- عند توقف الضياع نحصل على SVM.
- الانحدار اللوغاريتمي logarithmic regression .
- انحدار –التربيعات–الصغرى The least-squares - regression.

يتم عادةً استخدام هذا التابع على النحو التالي:

- معيار L2: (معين rhombus).
- معيار L1: (كروي sphere).
- شبكة مرنة: <l1 - 1. (معين محدب convex rhombus).

يعمل مصنف SGD بشكل أكثر ذكاءً cunningly باستخدام الانحدار التدريجي: حيث يتم تحديد بعض التقريب الأولي ثم يتم إجراء إعادة الحساب لكل متجه من مجموعة التدريب [13,14,15,16]:

$$w \leftarrow w - \eta\left(\alpha\frac{\partial R(w)}{\partial w} + \frac{\partial L(w^T x_i + by_i)}{\partial w}\right) \quad (3)$$

تتم إعادة حساب b بطريقة مماثلة ولكن بدون تنظيم regularization. قد تكون بعض الدالات ثابتة وبعضها الآخر خاص special، يؤثر اختيار هذه الدالات على سرعة التقارب للحل الأمثل. نلاحظ أن هذا المصنف هو مصنف ثنائي للحصول على تصنيف متعدد المراحل multiclass يجمع بين عدة مصنفات ثنائية في مخطط واحد.

3-2-2- تصنيفات KNN

تعمل خوارزمية الجار الأقرب K -nearest neighbor على النحو التالي: يتم العثور على K جار بعد أن يتم تعيين النقطة إلى الصنف الأكثر شيوعًا بين الجيران [17].

من مزايا هذه الطريقة:

- تنفيذ بسيط وواضح.
- تعمل بسرعة كبيرة.
- يمكن التعلم على تنفيذها.

من العيوب:

- لا تقاوم التدخل.
- تحتاج إلى تخزين كامل عينة التدريب في الذاكرة.

3-2-3- تصنيفات شجرة اتخاذ القرار Decision Tree

تعد أبسط خوارزميات التصنيف المقدمة وهي عبارة عن شجرة اتخاذ قرار عادية [17، 18]، غالبًا ما تستخدم كمجموعة شجرية (تسمى الغابة العشوائية Random Forest). تطبق هذه الخوارزمية في العديد من مشاكل التصنيف باستثناء تصنيف النص.

3-2-4- تصنيفات Multinomial Naive Bayes

تعد طرق Naive Bayesian مجموعة من خوارزميات التعلم المضبوطة بناءً على تطبيق نظرية Bayesian (4) مع الافتراض "naive" لاستقلالية دالة الاحتمال لكل متغير. من خلال تحديد متغير تسمية الفئة ومتجه المتغيرات x، تتم إعادة كتابة نظرية Bayesian بالعلاقة التالية:

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots x_n \mid y)}{P(x_1, \ldots, x_n)}$$

(5) بالنظر إلى افتراضنا "naive" للاستقلال

$$P(x_i|y, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = P(x_i|y),$$

يمكن تبسيط التعبير الأصلي بالشكل التالي:

(6)

$$P(y \mid x_1, \ldots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y) \quad (7)$$

$$\hat{y} = \arg\max_y P(y) \prod_{i=1}^{n} P(x_i \mid y),$$

يمكننا تحقيق تكوينات مختلفة للمصنف من خلال تحديد مساحات احتمالية مختلفة. سنقوم بالنظر إلى إحدى هذه التكوينات وتسمى المصنف Naive Bayesian Multi-nominal ويمكن استخدامه على نطاق واسع لحل مشكلة تصنيف النص.

يتم تحديد التوزيع باستخدام المتجه $\theta_y = (\theta_{y1}, \ldots, \theta_{yn})$ لكل تسمية من الفئة y، n هي بُعد مساحة الميزة (أي حجم قاموسنا)، θ_{yi} هو الاحتمال $P(x_i|y)$ للكلمة i التي تظهر في النص وتنتمي إلى الفئة y. يتم في الواقع العملي استخدام بعض الانحرافات عن هذه الطريقة خاصة θ_y حيث يتم استبدالها بالقيمة التالية [19 ، 20 ، 21]:

(8)

$$\hat{\theta}_{yi} = \frac{N_{yi} + \alpha}{N_y + \alpha n}$$

حيث $N_{yi} = \sum_{x \in T} x_i$ هو عدد المرات، وتكون مساهمة ظهور i في النصوص المسماة y في مجموعة التدريب T، و $N_y = \sum_{i=1}^{|T|} N_{yi}$ هو العدد الإجمالي لجميع سمات الفئة y.

يمكن بارامتر التنعيم $\alpha \geq 0$ smoothing parameter من مراعاة احتمال عدم دخول كلمة خاصة بفئة معينة لسبب محدد إلى مجموعة التدريب وبفضل هذا البارامتر لن يكون احتمال ظهور هذه الكلمة صفراً. $\alpha = 1$ يسمى تجانس لابلاس، و $\alpha < 1$ يسمى تجانس Lidstone.

3-3- وصف بيئة الأداة Description of the tool environment

تم اختيار لغة Python لعملية البرمجة ومكتبات sklearn + numpy لإجراء الحسابات [22، 23، 24]، ولهذا الأسلوب عدة مزايا مهمة:

- تتميز لغة بايثون ببنية مبسطة مع الكتابة التلقائية مما يبسط بشكل كبير تطوير البرامج النصية ذات التوجه العلمي، وهي واحدة من اللغات الثلاث الأكثر شيوعًا لتحليل البيانات.
- تحتوي مكتبة sklearn على الكثير من خوارزميات التعلم الآلي المنفذة ولديها توثيق جيد مع الكثير من الأمثلة على استخدام الكود.
- تحتوي مكتبة sklearn على مجموعة مضمنة من البيانات المرجعية للتحقق من جودة تصنيف النص (ليس هناك حاجة لإعداد عينات للتدريب واختبار المستندات النصية).
- يتم تنفيذ numpy الغالب ضمن لغة C، وبالتالي فإن سرعة معالجة البيانات مع هذه المكتبات غالبًا ما تتجاوز سرعة معالجة البيانات باستخدام البرامج المكتوبة بلغة R أو C#.

4- النتائـــج والمناقشـــة

تم أخذ المستندات النصية مباشرة من مجموعة sklearn.datasets. تم تمثيل مجموعة التدريب بـ 8120 مستندًا نصيًا، ومجموعة الاختبار بـ 5405 مستند. تضمنت محاور الوثائق مايلي:

1. حالات الطوارئ ذات الطابع الطبيعي:

- حرائق.
- الزلازل.
- فيضانات.
- تسونامي Tsunami.

2. حالة طارئة من صنع الإنسانman-made nature :

- حوادث في منشآت الهندسة الهيدروليكية.
- حوادث في منشآت خطرة للحريق fire hazardous facilities.
- حوادث المواد المتفجرة.
- حوادث المواد الخطرة كيميائياً.
- حوادث الأجسام الخطرة بالإشعاع.

3. تحليل السلامة البيئية:

- ضجيج Noise.
- اهتزاز Vibration.
- الأشعة فوق البنفسجية.
- أشعة الليزر.
- الاشعاع الكهرومغناطيسي.
- المعالجة بالإشعاع Radiation curing .

تم عرض نتائج التجارب في الجداول من 1–5 يبين الجدول 1 التجارب الأولية.

الجدول 1. التجارب الأولية

Feature Space	Classifier	TSVD	Accuracy
CountVectorizer	SGD	tsvd_off	0.901202590194
CountVectorizer	DesizionTree	tsvd_off	0.629787234043
CountVectorizer	RandomForest	tsvd_off	0.620536540241
CountVectorizer	KNN	tsvd_off	0.73691026827
CountVectorizer	NaiveBayes	tsvd_off	0.85420906568
CountVectorizer	SGD	tsvd_on	0.849398704903
CountVectorizer	DesizionTree	tsvd_on	0.548751156337
CountVectorizer	RandomForest	tsvd_on	0.623311748381
CountVectorizer	KNN	tsvd_on	0.625161887142
HashingVectorizer	SGD	tsvd_off	0.90046253469
HashingVectorizer	DesizionTree	tsvd_off	0.632932469935
HashingVectorizer	RandomForest	tsvd_off	0.551156336725
HashingVectorizer	KNN	tsvd_off	0.73598519889
HashingVectorizer	SGD	tsvd_on	0.83829787234
HashingVectorizer	DesizionTree	tsvd_on	0.527474560592
HashingVectorizer	RandomForest	tsvd_on	0.621276595745
HashingVectorizer	KNN	tsvd_on	0.601665124884

بناءً على التجارب الأولية تم التخلص من مصنفين: شجرة القرار والغابة العشوائية، بالإضافة إلى استخدام Hashing Vectorizer والتي تتضمن تحديد المصطلحات terms المستخدمة بشكل متكرر frequently فقط. كما ذكرنا سابقًا يمكن أن يكون للكلمات التي تتواجد بشكل نادر جداً تأثير كبير على تصنيف النص (على سبيل المثال: في نصوص تحليل البيانات قد تقع مجموعة من أسماء matplotlib و R مرة واحدة فقط ولكنها تشير على الفور إلى التوجه العلمي للاختبار). أظهرت عمليات التنفيذ في المتوسط باستخدام ميزة تحويل مساحة TSVD نتائج أسوأ قليلاً (عند 250 =k)، ولكن من الضروري إيجاد القيمة المثلى لـ k مسبقًا وبعد ذلك سيكون من الممكن مقارنة جودة النموذج بالمساحتين المميزتين بالكامل.

يبين الجدول 2 الدقة المرتبطة بنتائج البحث عن بارامتر TSVD من أجل مصنفي SGD و SVM.

الجدول 2. نتائج البحث عن بارامتر TSVD

Classifier	TSVD	Accuracy
SGD	200	0.822201665125
SGD	250	0.849583718779
SVM	300	0.0738205365402
SGD	300	0.847548566142
SGD	350	0.850693802035
SGD	400	0.857354301573
SGD	450	0.857354301573
SGD	500	0.857724329325
SGD	550	0.865309898242
SGD	600	0.86308973173
SGD	650	0.870490286772
SGD	700	0.875670675301
SGD	750	0.872155411656
SGD	800	0.875485661425
SGD	850	0.873450508788
SGD	900	0.876225716929
SGD	950	0.874005550416

كما يتضح من الجدول تكون الدقة accuracy أخفض عند البحث عن البارامترات وذلك عند استخدام مجموعة أقل من الكلمات. تكون أهم ميزة لاستخدام TSVD هي تقليل أبعاد مساحة السمات مما يسمح بتوسيع مجموعة المؤهلات المطبقة على مهمة محددة. نلاحظ أيضاً أنه من أجل حالة SVM (مع توابع kernel مختلفة -rbf خطية ومتعددة الحدود، تم وضع المتوسط في الجدول) كانت الدقة منخفضة بشكل كبير وبالتالي لم يساعد هذا المصنف ضمن هذه الشروط. تم ملاحظة أيضاً أن زيادة المعامل k أدى إلى زيادة كبيرة في وقت التشغيل واستخدام ذاكرة الوصول العشوائي مما يجعل من المستحيل زيادة قيمة هذا البارامتر عند استخدام نظام حسابي يتكون من جهاز واحد فقط، وبالتالي تم ترك الأسلوب الكلاسيكي مع مجموعة من الكلمات. إضافة لذلك تم أجراء تجارب لتقدير جودة خوارزمية KNN عند k مختلفة كما هو مبين بالجدول 3:

الجدول 3. النتائج البحث عن بارامتر KNN

Number of neighbors	Accuracy
k=1	0.749676225717
k=3	0.73691026827
k=5	0.73691026827
k=7	0.735615171138
k=9	0.725994449584
k=13	0.726549491212
k=17	0.724329324699
k=21	0.720259019426
k=25	0.715818686401
k=40	0.701202590194
k=55	0.682886216466
k=70	0.664569842738
k=85	0.653469010176

بناءً على هذه التجربة تم تجاهل خوارزمية KNN حيث أظهرت نتائج أقل بكثير من SGD و Naive Bayes.

تم اختبار Naive Bayes لقيم مختلفة للبارامتر α كما هو مبين بالجدول 4.

الجدول 4. نتائج البحث عن بارامتر Naive Bayes

Alpha value	Accuracy
alpha=0.0	0.111008325624
alpha=0.1	0.905457909343
alpha=0.2	0.897132284921
alpha=0.3	0.890101757632
alpha=0.4	0.883996299722
alpha=0.5	0.878075855689
alpha=0.6	0.871600370028
alpha=0.7	0.867345050879
alpha=0.8	0.863644773358
alpha=0.9	0.859944495837
alpha=1.0	0.85420906568

تم بعد ذلك البحث في جميع البارامترات المحتملة لـ SGD وتم وضعها بالجدول 5.

الجدول 5. نتائج البحث عن معلمات SGD

Loss function	Penalty function	Accuracy
Hinge	l2	0.901202590194
Hinge	l1	0.588529139685
Hinge	Elasticnet	0.834227567068
Log	l2	0.842553191489
Log	l1	0.519703977798
Log	elasticnet	0.718963922294
modified_huber	l2	0.913968547641
modified_huber	l1	0.776503237743
modified_huber	elasticnet	0.876225716929
squared_hinge	l2	0.913968547641
squared_hinge	l1	0.775948196115
squared_hinge	elasticnet	0.876965772433

Perceptron	l2	0.866975023127
Perceptron	l1	0.13413506013
Perceptron	elasticnet	0.529324699352
squared_loss	l2	0.903792784459
squared_loss	l1	0.697132284921
squared_loss	elasticnet	0.841073080481
Huber	l2	0.883441258094
Huber	l1	0.0732654949121
Huber	elasticnet	0.457354301573
epsilon_insensitive	l2	0.902867715079
epsilon_insensitive	l1	0.57335800185
epsilon_insensitive	elasticnet	0.835892691952
squared_epsilon_insensitive	l2	0.913598519889
squared_epsilon_insensitive	l1	0.751341350601
squared_epsilon_insensitive	elasticnet	0.872525439408

يمكن من هذا الجدول استنتاج أن أفضل الطرق المدروسة هي مزيج من مجموعة عادية من الكلمات تعمل ككلمات قياسية من نصوص تمت إزالة كلمات التوقف منها وذلك عند استخدام المصنف SGD والذي يعتمد على تعديل Huber كـ دالة الخسارة و l2-norm كدالة أفضلية.

يمكن أن تطبق النتائج التي حصلنا عليه ليس فقط على اللغة الإنجليزية ولكن أيضًا للغة العربية حيث نقوم فقط باستبدال قائمة كلمات التوقف وخوارزميات التسوية.

كنتيجة للتجارب التي أجريت تم عرض أفضل جودة من خلال خوارزمية (SGD + مجموعة كاملة من الكلمات). قد يحكون من المفيد وضع تفسير لنتيجة البحث واعتماد SGDأفضل مصنف .

5- المراجـــــع

1. Mogotsi I. C. Christopher d. manning, prabhakar raghavan, and hinrich schütze: *Introduction to information retrieval //Information Retrieval*. – 2008. Access mode: http://nlp.stanford.edu/IR-book/pdf/18lsi.pdf
2. *Documentation and examples of work with sklearn library*. Access Mode: http://scikit-learn.org/stable/tutorial/
3. Machine learning algorithms. Access Mode: http://www.machinelearning.ru
4. Sebastiani F. *Machine learning in automated text categorization* //ACM computing surveys (CSUR). – 2002. – Т. 34. – №. 1. – С. 1-47.
5. Aizpurua, A., Harper, S., Vigo, M. *Exploring the relationship between web accessibility and user experience* (2016) International Journal of Human Computer Studies, 91, pp. 13-23.
6. Kulakov, D.B., Semenov, S.E., Kulakov, B.B., Shcherbachev, P.V., Tarasov, O.I. *Hydraulic Bipedal Robots Locomotion Mathematical Modeling* (2015) Procedia Engineering, 106, pp. 62-70.

7. Sakulin, S., Alfimtsev, A., Solovyev, D., Sokolov, D. *Web page interface optimization based on nature-inspired algorithms* (2018) International Journal of Swarm Intelligence Research, 9 (2), pp. 28-46.

8. Mayorova, V.I. *Concept of using innovative-educational university centers of space services as an innovation for space education* (2012) Proceedings of the International Astronautical Congress, IAC, 12, pp. 10045-10049.

9. Sun, Q., Li, H., Campillo, J., (...), Wang, C., Zhang, Q. A *Comprehensive Review of Smart Energy Meters in Intelligent Energy Networks* (2016) IEEE Internet of Things Journal 3(4), pp.464-479

10. Shen, J., Wei, X., Kraposhin, V.S., Vekshin, B.S. *High cold plastic deformation of die steel 4Kh5VF1S subjected to hardening and tempering* (2012) Metal Science and Heat Treatment, 53 (9-10), pp. 503-504.

11. Zubkov, N.N., Polyakov, A.F., Shekhter, Yu.L. *The hydraulic characteristics of porous materials for a system of transpiration cooling* (2010) High Temperature, 48 (2), pp. 231-237

12. Moness, M., Moustafa, A.M. A *Survey of Cyber-Physical Advances and Challenges of Wind Energy Conversion Systems: Prospects for Internet of Energy* (2016) IEEE Internet of Things Journal 3(2), pp.134-145

13. Ocaya, R.O., Terblans, J.J. *C-language package for standalone embedded atom method molecular dynamics simulations of fcc structures* (2010) Softwarex 5, pp.227-233 Open Access

14. Xu, J., Luo, X., Wang, G., Gilmore, H., Madabhushi, A. A *Deep Convolutional Neural Network for segmenting and classifying epithelial and stromal regions in histopathological images* (2016) Neurocomputing 191, pp.214-223

15. Jiao, Z., Gao, X., Wang, Y., Li, J. *A deep feature based framework for breast masses classification* (2016) Neurocomputing 197, pp.221-231

16. He, Y., Lei, J., Li, Y., Leung, C.H.C. *A framework of query expansion for image retrieval based on knowledge base and concept similarity* (2016) Neurocomputing 204, pp.26-32

17. Leng, B., Liu, Y., Yu, K., Zhang, X., Xiong, Z. *3D object understanding with 3D Convolutional Neural Networks* (2016) Information Sciences 366, pp.188-201

18. Xu, Y., Wang, H., Herrera, F. *A distance-based framework to deal with ordinal and additive inconsistencies for fuzzy reciprocal preference relations* (2016) Information Sciences 328, pp.189-205

19. Lu, Q., Zhou, W., Li, H. *A no-reference Image sharpness metric based on structural information using sparse representation* (2016) Information Sciences 369, pp.334-346

20. He, C., Hu, C., Zhang, W., Li, X. *A parallel primal-dual splitting method for image restoration* (2016) Information Sciences 358-359, pp.73-91

21. Vázquez de Parga, M., García, P., López, D. *A sufficient condition to polynomially compute a minimum separating DFA* (2016) Information Sciences 370-371, pp.204-220

22. Park, C., Kim, D., Oh, J., Yu, H. *Using user trust network to improve top-k recommendation* (2016) Information Sciences 374, pp.1339-1351

23. Menahem, E., Schclar, A., Rokach, L., Elovici, Y. XML-AD: *Detecting anomalous patterns in XML documents* (2016) Information Sciences 326, pp.71-88