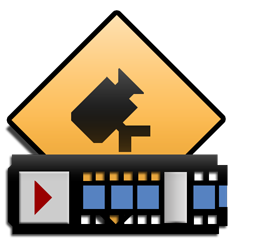
الجمهورية العربية السورية

جامعة دمشق

مادة التعلم التلقائي

|  |
| --- |
| **نظام لاقتراح الأفلام بحسب تقييمات المستخدم** |
| Movie Recommender System |



**بإشراف:**

د. جورج كراز د.عاصم القاسم

م. زينة الدلال

**تقدمة الطلاب:**

أيهم طالب خالد الرفاعي

عمار سعد الدين محمد فراس علايا

جدول المحتويات

[حساب منحنيات التعلم للخوارزميات المستخدمة 2](#_Toc325601435)

[منحني التعلم لخوارزمية الـ Collaborative Filteringبدلالة عدد عينات التدريب (m) 2](#_Toc325601436)

[مناقشة المخطط 3](#_Toc325601437)

[منحني التعلم لخوارزمية الـ Content-Based بدلالة عدد عينات التدريب (m) 3](#_Toc325601438)

[مناقشة المخطط 4](#_Toc325601439)

[تقييم النظام (System Evaluation) 4](#_Toc325601440)

# مقدمة

أصبح التسويق الالكتروني أحد ضروريات الحياة في مجتمعنا اليوم، فمئات الملايين من المستخدمين يتصلون بشبكة الإنترنت يومياً من جميع أنحاء العالم، ومن هذا المنطلق يمكننا استغلال تلك الفرصة وتحويل الانترنت إلى سوق مفتوح متنوع الجمهور لترويج السلع والخدمات المختلفة، وتحقيق ربحية عالية من وراء ذلك.

وكثير من الاشخاص يسئل أصدقائه ممن يشبهونه في ذوقهم عن رأيهم في منتجات لم يجربها من قبل , ومن هنا جاءت الحاجة إلى بناء موقع يقوم بهذه العملية بشكل آلي وتقديمها للزبائن .

وكان الهدف بناء نظام ذكي قادر على التنبؤ بسلوك الزبون و اهتماماته و تلبية رغباته عن طريق هذه الشبكة العنكبوتية.

# دراسة مرجعية

تقسم الخوارزميات المستخدمة من أجل الحصول على توقعات لمنتجات قد يرغب بها الزبون recommendations إلى العديد من الأنواع، ومن أهمها:

## Collaborative Filtering:

تعتمد هذه الطريقة على بعض قوانين الإحصاء الرياضي، وبإجراء عمليات حسابية يتم إيجاد التقارب بين المستخدم المطلوب حساب الاقتراحات له وبقية المستخدمين. للحصول على مجموعة من المستخدمين المتقاربين معه بشكل كبير.

أي يمكن الحصول على اقتراحات لمستخدم بحسب ذوق هذا المستخدم من خلال تحليل بياناته ( تقييمه للأفلام ) بهدف فهم ميول المستخدم لتقديم منتجات تتناسب مع وأهوائه، وذلك حسب الاشخاص المقاربين معه.

## Content-Based Filtering:

تختلف هذه الخوارزمية عن الخوارزمية السابقة، حيث أن الخوارزمية السابقة تقوم بتحليل بيانات المستخدم لفهم ميوله وسلوكه أي لا تدرس تقارب المستخدم مع بقية المستخدمين. بينما تقوم هذه الخوارزمية بدراسة تقارب الأفلام التي قام المستخدم بتقييمها مع بقية الأفلام.

ويتم تقديم كاقتراحات فقط الأفلام التي لها ارتباط عالي بالأفلام التي قام المستخدم بتقييمها.  
  
 وتعطي هذه الخوارزمية نتائج جيدة جداً كونها تعطي اقتراحات من نفس المجال التي قام المستخدم بتقييمها.

## A hybrid approach:

وهي خوارزمية هجينة بين الخوارزميتين content-based filtering and collaborative filtering.

تعتمد على السمات المستنتجة من المستخدمين من جهة وعلى السمات المستنتجة من المنتجات من جهة أخرى.

## Rule-Base Filtering:

وهنا تعتمد عملية التوقع على عمليات ربط لإيجاد مجموعات مرتبطة مع بعضها، فمثلاً قد نوجد مجموعة مرتبطة من المنتجات كما في المثال التالي:

إذا اشترى الزبون المنتج 1 والمنتج 2 فإنه سيشتري المنتج 3

وهذه العملية تفيد بأنه في الزيارة التالية لهذا المستخدم للموقع نقوم بإعطائه المنتج الثالث كـ recommendation.

# تصميم النظام

## Collaborative Filtering:

* تمت برمجة هذه الخوارزمية وتجريبها على البيانات التي قمنا بتجميعها، ومن ثم استعملنا أشعة السمات X و من أجل حساب الـRecommendations لكل مستخدم.

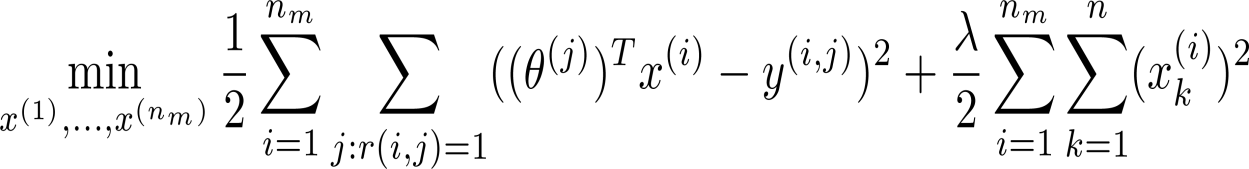
### تفاصيل الخوارزمية:

* تم استخدام خوارزمية Gradient Descent للتدريب.

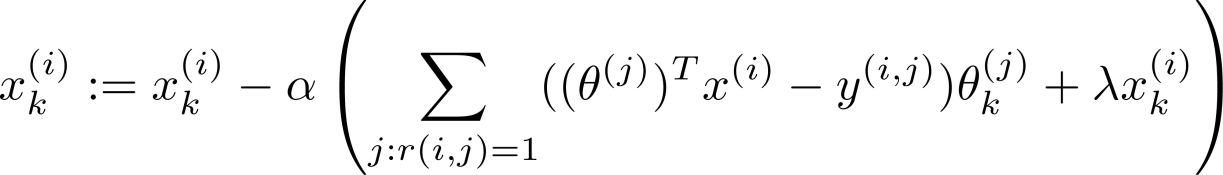
الدخل : مجموعة تقييمات الأفلام من المستخدمين

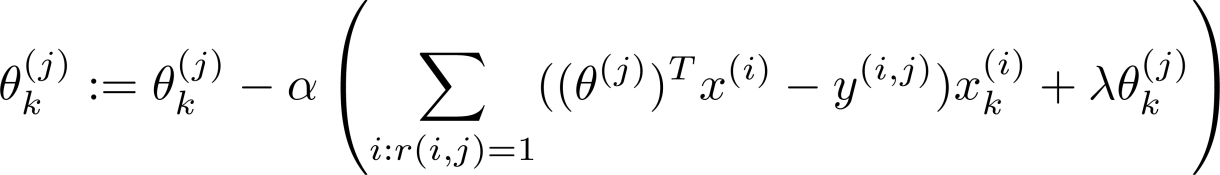
الخرج : أشعة السمات لكل فيلم ولكل مستخدم.

* الهدف استخدام الخرج السابق لإيجاد قيم التقييم المتوقعة لكل مستخدم مع كل فيلم لم يقم بتقييمه سابقاً , ومن ثم ترتيب قيم التوقع السابقة ترتيب تنازلي وعرضها كنصائح  **.**
  + حساب الخطأ :



* + حساب السمات باستخدام GD :





حيث قمنا بتجربة الخوارزمية السابقة على عدد من السمات، حتى توصلنا للعدد 10 للسمات وقمنا بالتثبيت عليه.

### طريقة حساب الـRecommendation للمستخدم الواحد:

- قمنا بحساب الـExpected Rating من أجل المستخدم الواحد على الفيلم الواحد بضرب شعاع سمات الفيلم الذي نريد اقتراحه للمستخدم بشعاع السمات الخاص بهذا المستخدم:

Expected Rating =

ومن ثم قمنا بترتيب هذه الـExpected Ratings لهذا المستخدم وقمنا بعرضها بترتيب تنازلي للقيم المتوقعة.

## Content-Based Filtering:

* تمت برمجة هذه الخوارزمية وتجريبها على البيانات التي قمنا بتجميعها، ومن ثم استعملنا أشعة السمات X و من أجل حساب الـRecommendations لكل مستخدم.

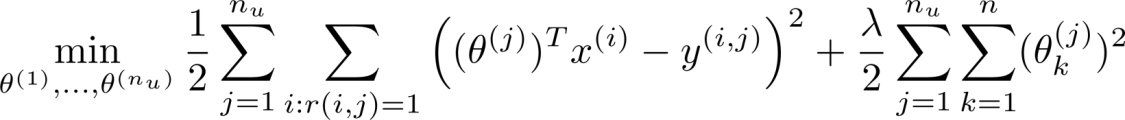
### تفاصيل الخوارزمية:

* تم استخدام خوارزمية Gradient Descent للتدريب.

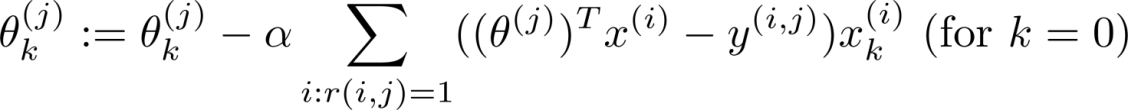
الدخل : مجموعة تقييمات الأفلام من المستخدمين بالإضافة إلى سمات كل فيلم.

الخرج : أشعة السمات لكل مستخدم.

* الهدف استخدام الخرج السابق لإيجاد قيم التقييم المتوقعة لكل مستخدم مع كل فيلم لم يقم بتقييمه سابقاً , ومن ثم ترتيب قيم التوقع السابقة ترتيب تنازلي وعرضها كنصائح  **.**
  + حساب الخطأ :



* + حساب السمات باستخدام GD :



* حيث تم تثبيت عدد السمات المخصصة لكل فيلم على العدد 24 سمة، حيث تمثل أول 21 قيمة منها قيم الانتماء للـGenres، وتحوي بعدها قيمة الـRelease للفيلم والـRevenue والAverage Rating .

### طريقة حساب الـRecommendation للمستخدم الواحد:

- قمنا بحساب الـExpected Rating من أجل المستخدم الواحد على الفيلم الواحد بضرب شعاع سمات الفيلم الذي نريد اقتراحه للمستخدم بشعاع السمات الخاص بهذا المستخدم:

Expected Rating =

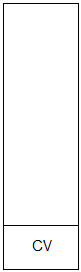
ومن ثم قمنا بترتيب هذه الـExpected Ratings لهذا المستخدم وقمنا بعرضها بترتيب تنازلي للقيم المتوقعة.

## A hybrid approach:

## حساب منحنيات التعلم للخوارزميات المستخدمة

فيما يلي رسم منحنيات التعلم للخوارزميات المستخدمة في تحقيق النظام أي رسم قيم منحنيي الخطأ في التدريب (training error) و الخطأ في الاختبار (test error) بدلالة حجم المعطيات المستعملة في التدريب، حيث تم أخذ قسم من المعطيات المتوفرة في قاعدة المعطيات (ratings) و تثبيتها للاختبار عليها، في حين يتم في كل مرة استعمال حجم متزايد من المعطيات للتدريب عليها، و من ثم تم مناقشة المخططات الناتجة.

ملاحظة: يتم أولاً خلط المعطيات المتوفرة في النظام بشكل عشوائي قبل أخذ القسم الخاص بالاختبار و ذلك كي لا تأخذ معطيات التقييم الكاملة لمستخدم ما إلى الاختبار دون أن تبقى له تقييمات ليتم التدريب عليها.

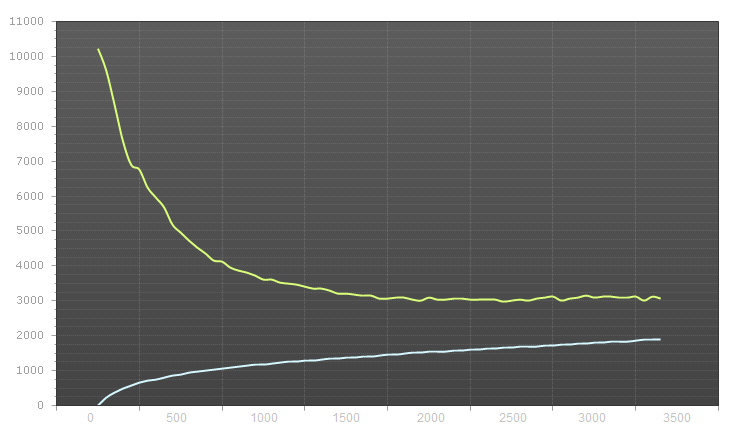


m

حيث m هو حجم المعطيات المستعملة للتدريب، و CV هو المجموعة المستعملة للاختبار.

### منحني التعلم لخوارزمية الـ Collaborative Filteringبدلالة عدد عينات التدريب (m)

تم خلط المعطيات (التي عددها 4792 تقييم)و أخذ حوالي 30% (أي 1437 عينة)من المعطيات لتكوِّن مجموعة الاختبار و من ثم تم أخذ مجموعات متزايدة للتدريب عليها و حساب قيمة الخطأ فيها و من ثم استعمال ناتج التدريب في حساب الخطأ على مجموعة الاختبار، و كان المخطط الناتج هو في الشكل التالي:



حيث:

المحور العمودي يمثل قيمة الخطأ.  
المحور الأفقي يمثل عدد عينات التدريب المستخدم.  
المنحني الأخضر هو منحني الخطأ لعينات الاختبار.  
المنحني الأبيض هو منحني الخطأ لعينات التدريب.

#### مناقشة المخطط

نلاحظ من المخطط أنه لدينا حالة high variance أي نحن في حالة overfitting و ذلك لأن الخطأ في التدريب صغير بالنسبة للخطأ في الاختبار أي هناك فجوة واضحة بين منحنيي الخطأين، و ذلك الخوارزمية تكون قد تعلمت فقط عينات التدريب.

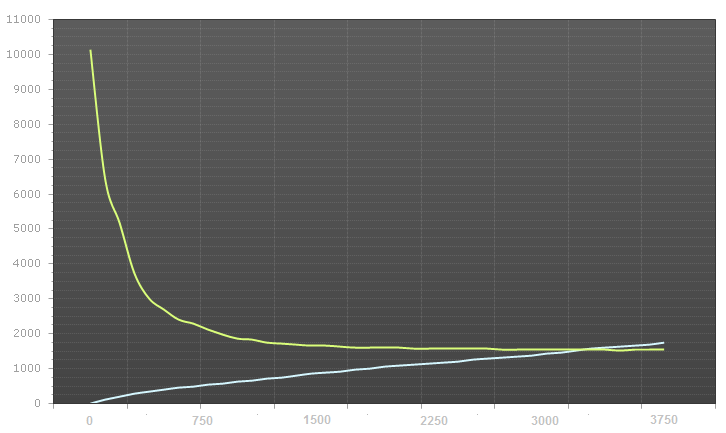
و لحل هذه المشكلة عادة نقوم بإحدى الأمور التالية التي قد تساعد على حلها:

1. الحصول على مجموعة معطيات أكبر
2. تقليل عدد السمات (features) (و هو حالياً 10).
3. زيادة معامل الـRegularization (λ).

ملاحظة: لم نتمكن من تجريب قيم مختلفة للمعاملات السابقة، و ذلك لأن كل تنفيذ لحساب منحني التعلم يأخذ حوالي أربع ساعات، و ذلك بزيادة عدد العينات بمقدار 75 عينة كفرق بين كل دورة و أخرى، أما لو كنا سننفذه بفرق عينة واحدة في كل دورة فذلك سوف يستهلك وقتاً أكثر من ذلك بكثير.

### منحني التعلم لخوارزمية الـ Content-Based بدلالة عدد عينات التدريب (m)

بنفس الطريقة السابقة تماماً يكون المخطط الناتج لخوارزمية الـContent-Based هو بالشكل:



حيث:

المحور العمودي يمثل قيمة الخطأ.  
المحور الأفقي يمثل عدد عينات التدريب المستخدم.  
المنحني الأخضر هو منحني الخطأ لعينات الاختبار.  
المنحني الأبيض هو منحني الخطأ لعينات التدريب.

#### مناقشة المخطط

نلاحظ هنا أن المخطط جيد جيداً حيث أن الخطأ يتقاطع في التدريب و الاختبار و بقيمة منخفضة نسبياً، و بالتالي نحن لسنا بحاجة لتعديل معامل الـRegularization، كما أننا لسنا بحاجة إلى الحصول إلى عدد أكبر من المعطيات لأن زيادتها لن تفيد في شيء، بالإضافة إلى أن عدد السمات في هذه الطريقة ثابت و هو 24.

ملاحظة: الطريقة الثالثة (The Hybrid Approach) لا تمتلك منحنيات خطأ و ذلك لأنها تُحسب باستخدام طريقة الـNormal Equasions، و إنما هي تقدم اقتراحات للأفلام و تحتاج طرق خاصة لتقييم أدائها.

# تقييم النظام (System Evaluation)

من الممكن تقسيم طرق تقييم أنظمة الاقتراحات (Recommender Systems) إلى بابين رئيسيين[2]:

* التقييم بشكل On-line: و هو الشكل البديهي لتقييم معيار نجاح النظام في إعطاء الاقتراحات المناسبة للمستخدمين و ذلك عن طريق بناء نظام كامل، و مراقبة سلوك المستخدمين لأخذ تغذية راجعة منهم، و ذلك إما بشكل مباشر عن طريق أخذ تقييم المستخدم للنتائج التي تُعرض لهم، أو بشكل غير مباشر مثل أن يقوم المستخدم بالضغط على الفلم المُقترح و استعراضه مما يدل على اهتمامه به، أو بأن يتجاهل المستخدم للفلم المُقترح مما يدل على فشل عملية الاقتراح، و هذه الطريقة تكون عادةً هي الأدق و لكن ليس من السهل دائماً تحقيقها لأنها تعتمد على وجود مستخدمين طيلة فترة الاختبار التي قد تأخذ وقتاً طويلاً.
* التقييم بشكل Off-line: حيث لا يُعتمد في هذه الطريقة على التغذية الراجعة من المستخدمين، وإنما يتم الاعتماد بشكل أساسي على المعطيات المجموعة، و غالباً ما يتم تقسيم المعطيات إلى مجموعة تدريب و مجموعة. و لكن هذه الطريقة بشكلها العادي من الغير ممكن أن تعطينا فكرة عن الخطأ الحقيقي الذي يعاني منه النظام، و إنما تعطينا الخطأ الظاهري فقط، و للحصول على الخطأ الحقيقي في أي مسألة فلا بد من الحصول على المعطيات الكاملة التي ممكن أن تتواجد في فضاء المسألة و التدريب عليها و هذا أمر شبه مستحيل في كثير من الأحيان، و لذلك من الممكن استخدام بعض الطرق التي تعطي خطأ قريب من الخطأ الحقيقي و ذلك باعتماد طرق مختلفة في تقسيم المعطيات إلى مجموعة تدريب و مجموعة اختبار، و من هذه الطرق:
  + طريقة K-Fold التي تعتمد على تقسيم المعطيات إلى K مجموعة و المرور على المعطيات K مرة بحيث يتم اختيار أحد هذه المجموعات للاختبار و التدريب على باقي المجموعات، و أخذ وسطي الأخطاء الناتجة عن كل مرور، و ميزة هذه الطريقة هي أنه كأننا نقوم بالتدريب على كافة المعطيات المتوفرة و الاختبار عليها كلها أيضاً، أي أننا لا نخسر من حجم المعطيات بعملية التقسيم.
  + طريقة Leave-one-out: و هي حالة خاصة من الطريقة السابقة بحيث يكون حجم المعطيات في كل مجموعة هو عينة واحدة. و هذه الطريقة تعطي تقييم أفضل للخطأ و لكنها ذات تعقيد أكبر.
  + طريقة الـBootStraping: التي تعتمد على تشكيل مجموعة جديدة للتدريب عدد العينات فيها من عدد عينات المعطيات الكلية، و ذلك عن طريق اختيار عناصر عشوائية من المعطيات الكلية و مع السماح بالتكرار، و العناصر التي لم يتم اختيارها هي التي تشكل مجموعة الاختبار، و نعيد العملية عدداً من المرات (العدد المُقترح 200 دورة) و نحسب متوسط الأخطاء الناتجة في الدورات، و الخطأ الناتج يُدعى بـe0.

و قد تم اختيار الطريقة K-Fold من الطرق السابقة و ذلك لأنها الأقل تعقيداً من حيث زمن التنفيذ، و قد تم تطبيقها على الطريقتين Collaborative Filtering و Content-Based و كانت نتائج الخطأ هي 1966.61 و 866.98على الترتيب مما يدل على أن تنفيذ الطريقة Content-Based في حالتنا هذه أفضل من طريقة الـCollaborative Filtering.

أما الطريقة الثالثة (Hybrid Approach) فإن المقالة [1] التي تم أخذها منها كان التقييم فيها بواسطة مقارنة المقترحات التي يقدمها النظام مع المقترحات التي يقدمها الموقع IMDB باستخدام نفس المستخدمين و تقييماتهم و نظراً لأننا جمعنا المعطيات من أشخاص من كليتنا فكان من الصعب تقييم هذه الطريقة بنفس الطريقة التي اتُّبعت في المقالة، و لكن تنص المقالة على أن نتائج هذه الطريقة هي أفضل من نتائج طريقة الـContent-Based.

# آفاق مستقبلية

* جعل الموقع متاحاً لكافة المستخدمين و ذلك برفعه على الشبكة العنكبوتية.
* بالإضافة إلى نظام الاقتراحات الذي كنا نعمل عليه فقد كنا نعمل على إضافة ميزة إلى الموقع و هي توقع أرباح الأفلام التي لما يتم إصدارها بعد، و لكن لم نستطع الانتهاء من تصميم و تقييم هذه الميزة بشكل مناسب، لذلك نحتاج إلى دراسة إضافية لتحقيقها.
* تقييم نظامنا بالطرق الـonline المذكورة مما يساهم في تقييمه بشكل مناسب أكثر و تعديل خواص خوارزمياته، مما يسمح بتحسين الأداء.

# خاتمة

لقد تم تصميم و اختبار موقع لاقتراح الأفلام المناسبة للمستخدمين اعتماداً على تقييماتهم لعدد من الأفلام المتوفرة في النظام، و تم استخدام ثلاث خوارزميات و مقارنتها في تنفيذ هذا النظام و تم الاستنتاج أن طريقة الـContent-Based أفضل من طريقة الCollaborative Filtering، أما الطريقة الثالثة (Hybrid Approach) فلم نقم بتقييمها و لكن تدل الدراسات على أنها أفضل الطرق.

# مراجع

[1] Feature Weighting in Content Based Recommendation System Using Social Network Analysis

Souvik Debnath, Niloy Ganguly,Pabitra Mitra

April 21-25, 2008 · Beijing, China

[2]An on-line evaluation framework for recommender systems  
C. Hayes1, P. Massa, P. Avesani, and P. Cunningham1

Trinity College Dublin, Ireland

*f*cunningham,cchayes*g*@cs.tcd.ie

ITC-IRST, Via Sommarive 18 - Loc. Pant`e, I-38050 Povo, Trento, Italy

*f*avesani,massa*g*@irst.itc.it