8823 – 7013M

Markscheme

November 2023

Computer science

Higher level

Paper 3

11 pages

© International Baccalaureate Organization 2023

All rights reserved. No part of this product may be reproduced in any form or by any electronic or mechanical means, including information storage and retrieval systems, without the prior written permission from the IB. Additionally, the license tied with this product prohibits use of any selected files or extracts from this product. Use by third parties, including but not limited to publishers, private teachers, tutoring or study services, preparatory schools, vendors operating curriculum mapping services or teacher resource digital platforms and app developers, whether fee-covered or not, is prohibited and is a criminal offense.

More information on how to request written permission in the form of a license can be obtained from https://ibo.org/become-an-ib-school/ib-publishing/licensing/applying-for-a-license/.

© Organisation du Baccalauréat International 2023

Tous droits réservés. Aucune partie de ce produit ne peut être reproduite sous quelque forme ni par quelque moyen que ce soit, électronique ou mécanique, y compris des systèmes de stockage et de récupération d’informations, sans l’autorisation écrite préalable de l’IB. De plus, la licence associée à ce produit interdit toute utilisation de tout fichier ou extrait sélectionné dans ce produit. L’utilisation par des tiers, y compris, sans toutefois s’y limiter, des éditeurs, des professeurs particuliers, des services de tutorat ou d’aide aux études, des établissements de préparation à l’enseignement supérieur, des fournisseurs de services de planification des programmes d’études, des gestionnaires de plateformes pédagogiques en ligne, et des développeurs d’applications, moyennant paiement ou non, est interdite et constitue une infraction pénale.

Pour plus d’informations sur la procédure à suivre pour obtenir une autorisation écrite sous la forme d’une licence, rendez-vous à l’adresse https://ibo.org/become-an-ib-school/ib-publishing/licensing/applying-for-a-license/.

© Organización del Bachillerato Internacional, 2023

Todos los derechos reservados. No se podrá reproducir ninguna parte de este producto de ninguna forma ni por ningún medio electrónico o mecánico, incluidos los sistemas de almacenamiento y recuperación de información, sin la previa autorización por escrito del IB. Además, la licencia vinculada a este producto prohíbe el uso de todo archivo o fragmento seleccionado de este producto. El uso por parte de terceros —lo que incluye, a título enunciativo, editoriales, profesores particulares, servicios de apoyo académico o ayuda para el estudio, colegios preparatorios, desarrolladores de aplicaciones y entidades que presten servicios de planificación curricular u ofrezcan recursos para docentes mediante plataformas digitales—, ya sea incluido en tasas o no, está prohibido y constituye un delito.

En este enlace encontrará más información sobre cómo solicitar una autorización por escrito en forma de licencia: https://ibo.org/become-an-ib-school/ib-publishing/licensing/applying-for-a-license/.

Subject details: **Computer science HL paper 3 markscheme**

Mark allocation

Candidates are required to answer **all** questions. Total 30 marks.

General

A markscheme often has more specific points worthy of a mark than the total allows. This is intentional. Do not award more than the maximum marks allowed for that part of a question.

When deciding upon alternative answers by candidates to those given in the markscheme, consider the following points:

- Each statement worth one point has a separate line and the end is signified by means of a semi-colon (;).
- An alternative answer or wording is indicated in the markscheme by a "/"; either wording can be accepted.
- Words in (…) in the markscheme are not necessary to gain the mark.
- If the candidate's answer has the same meaning or can be clearly interpreted as being the same as that in the markscheme then award the mark.
- Mark positively. Give candidates credit for what they have achieved and for what they have got correct, rather than penalizing them for what they have not achieved or what they have got wrong.
- Remember that many candidates are writing in a second language; be forgiving of minor linguistic slips. In this subject effective communication is more important than grammatical accuracy.
- Occasionally, a part of a question may require a calculation whose answer is required for subsequent parts. If an error is made in the first part then it should be penalized. However, if the incorrect answer is used correctly in subsequent parts then **follow through** marks should be awarded. Indicate this with "**FT**".
- Question 4 is marked against markbands. The markbands represent a single holistic criterion applied to the piece of work. Each markband level descriptor corresponds to a number of marks. When assessing with markbands, a "best fit" approach is used, with markers making a judgment about which particular mark to award from the possible range for each level descriptor, according to how well the candidate's work fits that descriptor.

General guidance

Issue	Guidance
Answering more than the quantity of responses prescribed in the questions	• In the case of an "identify" question read all answers and mark positively up to the maximum marks. Disregard incorrect answers. • In the case of a "describe" question, which asks for a certain number of facts *eg* "describe two kinds", mark the first two correct answers. This could include two descriptions, one description and one identification, or two identifications. • In the case of an "explain" question, which asks for a specified number of explanations *eg* "explain two reasons …", mark the first two correct answers. This could include two full explanations, one explanation, one partial explanation *etc*.

1. (a) *Award* ***[2 max]***
Cloud storage/storage services;
Virtual machines;
Processing / computation;
Primary Memory;
(Virtual) networks/Content Delivery Networks (CDN)/subnets;
Virtual Private Network (VPN);
Load balancing;
Firewalls;
Security (services/certification)/encryption;
Database (services);
Monitoring tools/Log management/analytics;
Integrated Development Environments (IDEs);
Software Development Kits (SDKs);
Application Programming Interfaces (APIs);
Technical support/consulting/compliance/certification;
Third party software (integrated into the IaaS platform);
Template (e.g. VM Image, infrastructure, configuration management);

(b) *Award* ***[2 max]***
A hyperparameter is a parameter that the model doesn't learn by itself;
And instead, it is provided to the model before it starts learning;

A hyperparameter is used to control the learning process;
Often with a trial-and-error approach;

A value that is set before the learning process begins;
And altered in the search for an optimal set/during tuning;

Note: Award a mark for a practical example for k-NN (e.g. k) or Matrix factorization (e.g. number of latent factors) and learning rate or epochs for NNs.

2. (a) *Award **[4 max]***
*Award **[1]** for tracking*
*Award **[1]** for profiling*
*Award **[1]** for identification*
*Award **[1]** De-anonymising data*
*Award **[1]** for predictive inferences*
*Award **[1]** for security*
*Award **[1]** for ethics/legislation*

Tracking: <u>Data gathering</u> and tracking techniques (e.g. cookies, user-agent string, keylogging) can be employed <u>without user awareness</u>;
Profiling: Implicit data gathering can create distinct profiles of users' habits, preferences, and behaviours (e.g. purchase history/content views profile);
Identification: These profiles can be used to identify individuals (especially when combined with other available data);
De-anonymisation: Data analysis techniques can correlate anonymous data with other datasets to de-anonymize individuals/companies should employ anonymizing techniques;
Predictive Inferences: User's personal characteristics, such as age, gender, political views, or interests can be inferred (revealing sensitive data);
Security: Implicit data, when linked to user profiles, increases the risk of data breaches/shared with third parties increases risk
Ethics/Legislation: Mandates certain levels of transparency about what data is collection/recommends users have the right to refuse implicit data collection.

(b) *Award **[4 max]***
*Award **[1]** for definition*
*Award **[1]** for diversification*
*Award **[1]** for balancing approaches*
*Award **[1]** for re-ranking strategies*
*Award **[1]** implicit data*
*Award **[1]** for customization,*
*Award **[1]** for fairness*
*Award **[1]** for evaluation*

Definition: Popular content is recommended frequently while less popular, niche or new content, is recommended rarely;
Diversification: Recommend a mix of popular and long-tail/add randomness;
Balancing approaches: hybrid approach combining collaborative filtering with content-based filtering;
Balancing approaches: Incorporating techniques (e.g. matrix factorization) not solely dependent on item popularity;
Balancing approaches: Apply regularization to address the long-tail preferences in factorization models/smooth the model or impose constraints;
Re-ranking strategies: Use standard approaches, then apply a re-ranking/post-step process;
Implicit data: Include implicit user feedback (like clicks, views, purchase history);
Customization: Users can set preferences or filters to control the balance of popular vs niche;
Fairness: Ensure the algorithm accounts for fairness, providing equitable visibility to items from different categories, creators, or providers;
Evaluation: Regularly evaluating the system with metrics that assess diversity, novelty, and coverage, not just accuracy and click-through rates;
Evaluation: Broadness of appeal check;

Note: Some answers could fall under several categories (e.g. adding a percentage of new content/artists to the recommendations could be diversification, balanced approaches, re-ranking, or fairness. Choose an empty category where possible.

3. *Award **[6 max]***
*Award **[1]** for initialize*
*Award **[1]** for factorize*
*Award **[2 max]** for model training*
*Award **[2 max]** for Learn Latent Features*
*Award **[1]** for prediction*
*Award **[1]** for ranking*
*Award **[1]** for recommendation*

Initialize: Set all items in the user matrix, U, and item matric, I, to random numbers/accept alternative initialisation approaches (e.g. SVD);
Factorization: Decompose the original user-item interaction matrix into two lower-dimensional latent matrices;
Model Training: Minimize the difference between the observed and predicted ratings/ minimize the cost function by comparing similarity between new and actual UI matrix;
Model Training: Use regularization to prevent overfitting;
Learn Latent Features: Iteratively adjust latent factors to minimize the reconstruction error/apply an optimization algorithm to adjust the values in the user matrix, U, and item matrix, I, a little at a time (e.g. gradient descent);
Learn Latent Features: Ongoing error monitoring during optimization to ensure accuracy and guide adjustments;
Learn Latent Features: Keep repeating until the 'cost function' is close to zero/end the optimization algorithm at a predetermined point;
Learn Latent Features: Multiply the user matrix by the items matrix to create a new UI matrix;
Prediction: Calculate missing entries in the matrix to predict user ratings for unrated items;
Ranking: Rank items for each user based on predicted ratings;
*Recommendation: R*ecommend the top-rated items/top of the tail;

Note: Accept diagrams that explain the steps for matrix factorisation

4. *Award* ***[12 max]***
Candidates may discuss some of the following points:

k-NN

- k-NN is one of the simplest forms of machine learning algorithms.
- k-NN can be used as a supervised learning algorithm and therefore needs a dataset for training and testing.
- k in k-NN represents the number of the nearest neighbours used to classify new data points.
- Choosing the right value of K is called parameter tuning and is necessary for better recommendations.
- The similarity measure (e.g. Euclidean, Manhattan, Minkowski, cosine similarity, etc.) used in k-NN affects how well the algorithm identifies users with similar preferences. These can be changed and evaluated like hyperparameter tuning.
- k-NN can be adapted to different contexts, for instance, by using different distance metrics suitable for the specific nature of the recommender system.

Training and Testing Phase

- Since k-NN is a lazy learning algorithm, it doesn’t undergo a conventional training phase like other machine learning models. Rather than learning a function from the training data, k-NN uses the entire dataset for making predictions during the testing phase.
- Despite being a lazy learner, validation steps like k-fold cross-validation are essential in k-NN to determine the optimal number of neighbours (k) and to assess the model's effectiveness and generalizability on unseen data.
- Testing in k-NN involves using the model to predict outcomes for data not used during the validation phase. This step assesses the quality of the predictions and the model's ability to generalize from the training data to new, unseen data.

K-NN choice of algorithm

- k-NN is a good choice if you have a small dataset. However, k-NN relies heavily on historical data to make predictions or recommendations. In the absence of sufficient data, particularly for new users or items, k-NN struggles to identify 'neighbours' and make accurate predictions.
- k-NN works well when the data is noise free.
- k-NN only works well when the data is properly labelled.
- k-NN is non-parametric so lacks statistical power. Thus, doesn't make any assumptions about the underlying data distribution.
- K-NN has a tendency towards popularity bias.
- The accuracy and precision of k-NN in a recommender system can be sensitive to sparse data and the curse of dimensionality, potentially limiting its effectiveness. Techniques like dimensionality reduction or embedding methods might be necessary to address this.
- Data quality and preprocessing steps (like normalization or handling missing values) are crucial for k-NN's performance in delivering accurate recommendations.
- k-NN's performance for a recommender system might suffer due to its sensitivity to noisy and irrelevant features unless feature selection is properly conducted.
- The algorithm may need to be adapted or extended (e.g. by weighting neighbours based on similarity) for better accuracy.
- k-NN can be computationally expensive for large datasets, impacting real-time recommendation scenarios. However, techniques like indexing (e.g., KD-trees, Ball trees) can speed up nearest-neighbour searches in practice.
- The k value must be chosen carefully for the data sample.
- A poorly chosen value for k may be misrepresentative the skill of the model, such as a score with a high variance (that may change a lot based on the data used to fit the model), or a high bias (such as an overestimate the skill of the model).
- Evaluation of k-NN-based recommender systems should involve relevant metrics like precision, recall, and F1-score,
- Cross-validation/K-fold cross validation (a resampling procedure used to evaluate machine learning models on a limited data sample) might be used.

Overfitting in k-NN

- The value of k plays a crucial role in determining whether the k-NN algorithm overfits or underfits the data.
- A very small value of k (such as k=1 or k=2) makes the algorithm extremely sensitive to noise in the training data, leading to overfitting.
- k-NN is particularly prone to overfitting in high-dimensional spaces (known as the curse of dimensionality).
- Overfitting in k-NN can also occur due to the presence of noise and outliers in the dataset.
- Overfitting can be exacerbated when irrelevant or less important features are included in the dataset, especially since k-NN treats all features equally.
- Strategies to reduce overfitting include dimensionality reduction, feature selection, noise filtering, and using robust distance metrics.

Second Evaluation of Training and Testing

- The traditional notions of training and testing don't apply to k-NN in the same way as they do to other algorithms. However, it is possible to apply a second evaluation procedure on a k-NN algorithm.
- A large percentage (approx. 75%) of the dataset is used for training.
- Validation (approx. 15% to 20%) is carried out while training occurs.
- Initial accuracy is evaluated and hyperparameters are fine-tuned.
- The validation data is not used to learn weights and biases.
- A small percentage (approx. 5% to 10%) of the dataset is used for testing.
- Having never seen the testing data, the model is free from any bias.

Evaluation

- The need to evaluate with relevant metrics (precision, recall, F1-score).
- Precision assesses the proportion of recommended items that are relevant to the user's interests out of all recommended items.
- Recall assesses the proportion of relevant items that are successfully recommended out of all the relevant items available.
- F1-score provides a harmonic mean of precision and recall, offering a single score that balances both concerns.
- Explaining cross-validation as a method for assessing generalizability.
- Discussion of the use of a test set to evaluate the final model performance.
- A/B testing – presenting two variants of the algorithm (A and B) to different user groups to determine which version yields better performance in terms of accuracy, engagement, click-through rates, or conversion rates.

Conclusion.
A final conclusion in which the candidate links together the various points.

Please see markband.

Marks	Level descriptor
No marks	• No knowledge or understanding of the relevant issues and concepts. • No use of appropriate terminology.
Basic 1–3 marks	• Minimal knowledge and understanding of the relevant issues or concepts. • Minimal use of appropriate terminology. • The answer may be little more than a list. • No reference is made to the information in the case study or independent research.
Adequate 4–6 marks	• A descriptive response with limited knowledge and/or understanding of the relevant issues or concepts. • A limited use of appropriate terminology. • There is limited evidence of analysis. • There is evidence that limited research has been undertaken.
Competent 7–9 marks	• A response with knowledge and understanding of the related issues and/or concepts. • A response that uses terminology appropriately in places. • There is some evidence of analysis. • There is evidence that research has been undertaken.
Proficient 10–12 marks	• A response with a detailed knowledge and clear understanding of the computer science. • A response that uses terminology appropriately throughout. • There is competent and balanced analysis. • Conclusions are drawn that are linked to the analysis. • There is clear evidence that extensive research has been undertaken.