

آخر تطورات أدوات الذكاء الاصطناعي التوليدية لدعم إمكانية النفاذ في مجال تطوير المنتجات

إياد أبو دوش
كلية الهندسة والعلوم التطبيقية
الجامعة الأمريكية في الكويت
idoush@auk.edu.kw
السالمية، الكويت

الملخص:

تعالج هذه الورقة الحاجة الملحة لتقييم نضج ومقاييس الأداء لأدوات الذكاء الاصطناعي التوليدي المخصصة لإمكانية النفاذ في مجال تطوير المنتجات. وتكمن المشكلة في هذا المجال في عدم وجود طرق موحدة لتقييم نضج أدوات الذكاء الاصطناعي التوليدي المخصصة لاحتياجات إمكانية النفاذ وغياب مقاييس الأداء المقبولة عالميًا لقياس فعاليتها. حيث يعيق هذا النقص تقدم ممارسات التصميم الشاملة ويحد من التأثير المحتمل لحلول إمكانية النفاذ التي تعتمد على الذكاء الاصطناعي. وتقتصر هذه الورقة إطارًا شاملاً لتقييم نضج أدوات الذكاء الاصطناعي المصممة خصيصًا لإمكانية النفاذ في تطوير المنتجات. لقد قمنا بتوضيح المعايير الحاسمة المتكاملة لهذا التقييم والتي تشمل جوانب مثل قابلية الاستخدام والموثوقية وقابلية التوسع والقدرة على التكيف مع احتياجات وسياقات المستخدم المتنوعة. ويهدف الحل المقترح إلى المساهمة بالمعرفة القيمة في المشهد المتطور لأدوات الذكاء الاصطناعي التوليدي المخصصة لتعزيز إمكانية النفاذ في مجال تطوير المنتجات. وتسعى أبحاثنا من خلال إنشاء نهج منظم لتقييم مستوى النضج والدعوة إلى مقاييس الأداء الموحدة إلى تمكين المطورين والمصممين وأصحاب المصلحة من اتخاذ قرارات مستنيرة بشأن تبني وتحسين حلول إمكانية النفاذ التي تعتمد على الذكاء الاصطناعي.

الكلمات الرئيسية: شات جي بي تي (ChatGPT) - مساعد الذكاء الاصطناعي (Copilot) - إمكانية النفاذ الرقمي - الذكاء الاصطناعي - إنشاء أكواد البرمجة - تطوير المنتجات القابلة للنفاذ.

1. المقدمة

لقد ركزت العديد من المساعي البحثية على تعزيز إنتاجية المطورين من خلال مناهج مختلفة مثل توليف الأكواد البرمجية والبحث عنها وأشكال أخرى من "أنظمة التوصية بالأكواد البرمجية" [1]. وتستفيد العديد من هذه المبادرات من قدرات الذكاء الاصطناعي وخاصة استخدام تقنيات التعلم العميق [2]. وقد حدث تطور كبير في هذا المجال في يونيو 2021 عندما قدمت (GitHub) و (OpenAI) مساعد الذكاء الاصطناعي (GitHub Copilot) "المبرمج المزدوج باستخدام الذكاء الاصطناعي" وهي أداة برمجية ثورية متوافقة مع العديد من بيئات التطوير المتكاملة. ويفضل نموذج (OpenAI Codex) الموسع الذي تم تدريبه على مجموعة كبيرة من أكواد (GitHub) البرمجية مفتوحة المصدر فقد حقق مساعد الذكاء الاصطناعي (Copilot) تفوقاً ملحوظاً في مجال اقتراح مقاطع الأكواد البرمجية المخصصة (code snippets) في لغات برمجة مختلفة. كما يوفر (ChatGPT) وهو أداة قوية أخرى في هذا المجال واجهة محادثة يمكن الاستفادة منها لتوليد الأكواد البرمجية وتطوير المنتجات المختلفة [3]. وتستفيد هذه الأدوات من مجموعات البيانات الضخمة لإنشاء نماذج توليدية بهدف إنتاج أكواد برمجية تسهل تطوير المنتجات القابلة للنفاذ. ورغم كل ماسبق فإن مدى نضج هذه الأدوات يبقى أمراً من الأهمية بمكان أخذه في عين الاعتبار.

يوفر مساعد الذكاء الاصطناعي (Copilot) للمستخدمين ثلاث وظائف أساسية: تحويل التعليقات إلى كود برمجي واقتراح الاختبارات التي تتطابق مع كود التنفيذ والملء التلقائي للكود المتكرر. وتبدأ وظيفة تحويل التعليقات إلى كود عندما يكتب المستخدم تعليقاً يصف المنطق الذي ينوي تنفيذه. وعلى الرغم من أنه يمكن طلب اقتراحات مساعد الذكاء الاصطناعي (Copilot) من خلال تعليقات باللغة الطبيعية فقط إلا أن النتائج المثلثة تتحقق عندما يكمل المستخدمون مدخلاتهم بأسماء ذات معنى لمعاملات الوظيفة (function parameters) والتعليقات الوصفية [2].

يمكن استخدام (ChatGPT) لتوليد الكود البرمجي من خلال الانخراط في محادثة معه. ويحتاج المستخدم هنا إلى التعبير بوضوح عن نية توليد الكود ثم تقديم السياق والتفاصيل ذات الصلة حول الكود المطلوب. ويجب أخيراً أن يتفاعل المستخدم مع (ChatGPT) من خلال سلسلة من "الموجهات" (prompts) والاستجابات لتحسين الكود المنتج [3].

تم في العمل [4] استكشاف استخدام نماذج اللغة الكبيرة (LLM) مثل (ChatGPT) لتحسين إمكانية النفاذ إلى الويب تلقائياً. حيث تقارن ورقة البحث فعالية (ChatGPT) في اكتشاف مشكلات إمكانية النفاذ وإصلاحها مع اختبار ذلك يدوياً في موقعين غير متوافقين مع معايير النفاذ. وتظهر النتائج أداءً واعداً من قبل (ChatGPT) في مجال تحسين إمكانية النفاذ إلى الويب وهو أمر حيوي لتلبية إرشادات النفاذ إلى محتوى الويب (WCAG 2.1) وإنشاء منصات أكثر شمولاً عبر الإنترنت. كما شهد مجال الذكاء الاصطناعي المزدهر استخداماً واسع النطاق للأدوات التي تهدف إلى تعزيز إمكانية النفاذ في مجال تطوير المنتجات.

يتناول هذا البحث عدة أسئلة رئيسية تشمل الطرق الفعالة لتقييم نضج أدوات الذكاء الاصطناعي التوليدي المخصصة لإمكانية النفاذ في مجال تطوير المنتجات ويوضح المعايير المتكاملة لهذا التقييم. ثم يتعمق البحث في مقاييس الأداء المتنوعة المستخدمة لقياس فعالية وكفاءة هذه الأدوات ويقترح تدابير محددة لتوحيد المعايير. ويهدف هذا البحث إلى تقديم معارف قيمة للمساهمة في المشهد المتطور لأدوات الذكاء الاصطناعي التوليدي المخصصة لتعزيز إمكانية النفاذ في تطوير المنتجات.

وتعمل المقاييس المختلفة كمعايير لتقييم فعالية وشمولية مخرجات أكواد أدوات الذكاء الاصطناعي التوليدي. ومن المقاييس التي تعتبر محورية للتقييم الشامل في هذا السياق نذكر وظائف الكود وبنية المرئية ومدى تعقيده واعتبار إمكانية استخدام أجهزة الإدخال المتعددة ومستوى خطورة مشكلات إمكانية النفاذ داخل الكود الناتج. ومن خلال استخدام هذه التدابير المتعددة الأوجه في عملية التقييم سيتمكن أصحاب المصلحة من اتخاذ قرارات مستنيرة بشأن اختيار وتحسين أدوات الذكاء الاصطناعي لإمكانية النفاذ في تطوير المنتجات. ويضمن هذا النهج الشامل تقديم تجارب رقمية شاملة تلبي الاحتياجات المتنوعة للمستخدمين ذوي القدرات المختلفة.

2. الذكاء الاصطناعي التوليدي لتحسين إمكانية النفاذ في أكواد البرمجة

لقد بشر استخدام الذكاء الاصطناعي في صناعة البرمجيات في السنوات الأخيرة بتطورات كبيرة في إمكانية النفاذ. ويبرز الذكاء الاصطناعي التوليدي بين هذه الابتكارات كقوة تحويلية وخاصة في مجال إمكانية النفاذ في أكواد البرمجة. ويستكشف هذا القسم المشهد المزدهر للذكاء الاصطناعي التوليدي وتأثيراته العميقة لتعزيز إمكانية النفاذ في تطوير المنتجات. لا يعمل الذكاء الاصطناعي التوليدي على تسريع عملية التطوير فحسب من خلال أتمتة إنشاء وتعديل هياكل أكواد البرمجة فهو يعزز أيضاً الشمولية من خلال تمكين المطورين من جميع مستويات المهارة من إنشاء برامج قابلة للنفاذ. وستتعمق في هذا السياق في أربع حالات مميزة حيث يثبت الذكاء الاصطناعي التوليدي فعاليته [5].

أ. إنشاء أكواد البرمجة لإمكانية النفاذ: تستخدم أدوات الذكاء الاصطناعي مجموعات بيانات واسعة النطاق لإنشاء أكواد برمجية تعزز إمكانية النفاذ في تطوير المنتجات. ومع ذلك فإن نضج هذه الأدوات لا يزال قيد التطور. ويكمن أحد التحديات الرئيسية في مدى قدرتها على وصف المحتوى بدقة من خلال أكواد البرمجة. ففي حين يمكن لهذه الأدوات اقتراح سمات إمكانية النفاذ إلا أن تطبيق هذه السمات بقيم دقيقة بناءً على المحتوى لا يزال يشكل عقبة ملموسة. وفيما يلي مثال لحالة استخدام موجه "أمر" لإنشاء كود قابل للنفاذ باستخدام مساعد الذكاء الاصطناعي (Copilot):

الموجه (الأمر): اعمل كمدرّب خبير في مجال إمكانية النفاذ لإنشاء جدول (HTML) قابل للنفاذ ويتوافق مع معايير النجاح (A) و(AA) الخاصة بإرشادات النفاذ إلى محتوى الويب (WCAG 2.1). يحتوي الجدول على 5 صفوف و3 أعمدة تحتوي على العناوين التالية: الاسم والبريد الإلكتروني

وتاريخ الميلاد. استخدم عند الإمكان عناصر (HTML) الدلالية للتأكد من إمكانية تشغيل الجدول باستخدام لوحة المفاتيح. واتبع أيضًا دليل ممارسات إنشاء تطبيقات الإنترنت الغنية القابلة للنفاز (ARIA) وأنماط التصميم ذات الصلة.

الكود المُنشأ: فيما يلي مثال لجدول قابل للنفاز مع العناوين والمحتوى المحددين:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Accessible Table Example</title>
</head>
<body>
  <table role="grid" aria-label="User Information">
    <thead>
      <tr>
        <th scope="col">Name</th>
        <th scope="col">Email</th>
        <th scope="col">Date of Birth</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>John Doe</td>
        <td><a
href="mailto:john@example.com">john@example.com</a></td>
        <td>1985-07-15</td>
      </tr>
      <tr>
        <td>Jane Smith</td>
        <td><a
href="mailto:jane@example.com">jane@example.com</a></td>
        <td>1990-03-22</td>
      </tr>
      <!--Add more rows as needed -->
    </tbody>
  </table>
</body>
</html>
```

ب. ملاحظات (التغذية الراجعة) إمكانية النفاذ: لا تزال موثوقية الملاحظات من أدوات الذكاء الاصطناعي تشكل مصدر قلق عند تقييم مشكلات إمكانية النفاذ في الكود. فقد تواجه هذه الأدوات على سبيل المثال صعوبة في تحديد ما إذا كانت القيم المعينة لسمات النفاذ تصف المحتوى بدقة. ويفرض هذا الغموض تحديات أمام المطورين الذين يعتمدون على الملاحظات التي يولدها الذكاء الاصطناعي لضمان الامتثال لإمكانية النفاذ.

الموجه (الأمر): اعمل كمدرّب خبير في مجال إمكانية النفاذ لتحديد مشكلات إمكانية النفاذ وكيفية إصلاحها في جدول (HTML) التالي وذلك لضمان توافق الكود مع معايير النجاح (A) و(AA) الخاصة بإرشادات النفاذ إلى محتوى الويب (WCAG 2.1). عندما يكون ذلك ممكناً، تحقق من استخدام عناصر (HTML) الدلالية للتأكد من إمكانية تشغيل الجدول باستخدام لوحة المفاتيح. تأكد أيضاً من أنه يتبع دليل ممارسات إنشاء تطبيقات الإنترنت الغنية القابلة للنفاذ (ARIA) وأنماط التصميم ذات الصلة. فيما يلي كود جدول (HTML):

```
<table>
  <tr>
    <td>Row 1, Column 1</td>
    <td>Row 1, Column 2</td>
  </tr>
  <tr>
    <td>Row 2, Column 1</td>
    <td>Row 2, Column 2</td>
  </tr>
  <tr>
    <td>Row 3, Column 1</td>
    <td>Row 3, Column 2</td>
  </tr>
</table>
```

الاستجابة: نعرض فيما يلي القسم الخاص بالتوصيات من الاستجابة التي تم توفيرها:

- أضف عنصر "تسمية توضيحية" <caption> لوصف غرض الجدول (على سبيل المثال، "بيانات المبيعات للربع الأول من عام 2024").
- إذا كان الجدول يحتوي على رؤوس (Headers) استخدم عناصر <th> لها.
- قدم نصًا بديلاً لأي محتوى غير نصي (إذا أمكن).
- اختبر التنقل باستخدام لوحة المفاتيح داخل الجدول.

أ. توصيات وإرشادات إصلاح مشكلات النفاذ: تعد القدرة على تقديم توصيات إصلاح المشكلات أحد الجوانب الجوهرية لاستخدام أدوات الذكاء الاصطناعي في تطوير المنتجات. فبالرغم من أنه بإمكان

هذه الأدوات تحديد مشكلات إمكانية النفاذ إلا أن قدرتها على توجيه المطورين لإصلاح هذه المشكلات تعد أمراً بالغ الأهمية.

ب. إنشاء حالات الاختبار: لا ينتظر من أدوات الذكاء الاصطناعي تحديد مشكلات إمكانية النفاذ فحسب بل أن تساهم أيضاً بنشاط في إنشاء أكواد وحالات اختبار قابلة للنفاذ. وينعكس نضج هذه الأدوات في قدرتها ليس فقط على الإشارة إلى المشكلات ولكن أيضاً اقتراح الحلول.

إن أحد التحديات الرئيسية في مجال استخدام أدوات الذكاء الاصطناعي لتطوير برمجيات المنتجات القابلة للنفاذ هو توظيف المحتوى وإنشاء أكواد البرمجة بطريقة تضمن التماسك والاتساق الدلالي والهيكلية. فقد تتمكن أداة الذكاء الاصطناعي على سبيل المثال من اقتراح سمات إمكانية النفاذ لعناصر (HTML) مثل (alt أو aria-label أو role) ولكنها قد لا تتمكن من تعيين قيم مناسبة لهذه السمات بناءً على محتوى وسياق صفحة الويب. كما قد لا تتمكن أداة الذكاء الاصطناعي من التحقق مما إذا كانت القيم التي يوفرها المطور دقيقة ووصفية بما يكفي للتكنولوجيا المساعدة لمساعدة المستخدمين ذوي الإعاقة. ومن هذا المنطلق فإنه لا ينبغي لأداة الذكاء الاصطناعي إنشاء الأكواد البرمجية فحسب بل يجب عليها أيضاً تقديم ملاحظات وتوصيات لإصلاح لمساعدة المطورين في إصلاح مشكلات إمكانية النفاذ وتحسين جودة أكواد البرمجة.

3. مؤشرات الأداء لإمكانية النفاذ إلى أكواد البرمجة

يمثل تقييم فعالية أدوات الذكاء الاصطناعي في ضمان إمكانية النفاذ إلى أكواد البرمجة تحدياً متعدد الأوجه. ويوجد حالياً نقص في المنهجيات الموحدة لتقييم إمكانية النفاذ إلى أكواد البرمجة مما يستلزم تحديد مؤشرات الأداء المناسبة. وتشمل هذه المؤشرات جوانب مختلفة مثل كمية وشدة أخطاء إمكانية النفاذ التي تم اكتشافها بواسطة الأداة وشمولية ودقة أكواد البرمجة التي أنتجتها الأداة ورضا وقابلية استخدام الأشخاص ذوي الإعاقة الذين يتفاعلون مع المنتج والالتزام بمعايير وإرشادات إمكانية النفاذ ذات الصلة. ومع ذلك فإنه من الضروري إدراك أن هذه المؤشرات يمكن أن تتأثر بعوامل مثل طبيعة المنتج والغرض منه وتعقيده بالإضافة إلى خصائص جمهوره المستهدف وسيناريوهات المستخدم. وبالتالي فإنه ينبغي على أداة الذكاء الاصطناعي الفعالة إثبات قدرتها على التكيف عبر سياقات متنوعة وتقديم رؤى قابلة للتنفيذ لكل من المطورين وأصحاب المصلحة.

إن هناك حاجة إلى نهج شامل في عملية تقييم أداء أدوات الذكاء الاصطناعي لتعزيز إمكانية النفاذ في تطوير المنتجات. ويتضمن هذا النهج النظر في مقاييس مختلفة لقياس فعالية الأداة. وقد تشمل هذه

المقاييس الدقيقة في تحديد مشكلات إمكانية النفاذ والدقة في اقتراح سمات إمكانية النفاذ ذات الصلة واستخدام قيم قائمة على السياق لضمان دقة المحتوى. كما تشمل العوامل الحاسمة لتقييم الأداة فعاليتها في تقديم توصيات لإجراء إصلاحات قابلة للتنفيذ وقدرتها على إنشاء كود وحالات اختبار قابلة للنفاذ إضافة إلى قدرتها على تقديم ملاحظات ثاقبة حول مستوى إمكانية النفاذ العام للكود. ويمكن لأصحاب المصلحة من خلال التقييم الشامل لهذه الجوانب اكتساب فهم أعمق لتأثير الأداة على تعزيز نفاذية الكود وضمان تلبية الاحتياجات المتنوعة للمستخدمين عبر سياقات وسيناريوهات مختلفة.

تشمل الأعمال السابقة في مجال تقييم الكود الناتج عن الذكاء الاصطناعي [6] الذي يقدم مقياس لتقييم جودة الكود الناتج بما يشمل أخطاء وقت التحويل البرمجي ووقت التشغيل والمخرجات الخاطئة وأسلوب الكود وقابلية الصيانة والكفاءة. ويقترح عمل آخر [7] استخدام عملية التحقق من صلاحية الكود وصحته وكفاءته من حيث تعقيد الوقت والمكان.

نقترح في هذا العمل مجموعة من التدابير لتقييم نفاذية الكود الذي تم توليده. ولم يسبق أن تقديم هذه التدابير في الأدبيات ذات الصلة. أولاً، يلعب الترتيب المرئي للكود دوراً حاسماً في تعزيز قابلية القراءة والفهم وخاصة للمطورين والأشخاص الذين يستخدمون برامج قراءة الشاشة. وبالتالي فإن على الأدوات إعطاء الأولوية لإنشاء كود بهيكل واضح ومنطقي مما يسهل التنقل فيه وفهمه. ثانياً، يساعد النظر في استخدام أجهزة الإدخال المتعددة على توسيع نطاق إمكانية النفاذ واستيعاب المستخدمين ذوي القدرات المختلفة. ومن هنا فإن على أدوات الذكاء الاصطناعي أن تنتج كوداً يدعم طرق إدخال متنوعة بما في ذلك التنقل عبر لوحة المفاتيح والأوامر الصوتية وعناصر التحكم بالإيماءات مما يعزز تجربة المستخدم الشاملة. وأخيراً يساعد تقييم مستوى شدة مشكلات إمكانية النفاذ في الكود المؤكد في تحديد الأولويات ومعالجة العوائق الهامة أمام النفاذ. حيث يجب تقييم الكود المنتج بواسطة أدوات الذكاء الاصطناعي من حيث شدة أوجه القصور في إمكانية النفاذ ومدى تمكين المطورين من تخصيص الموارد بشكل فعال وإعطاء الأولوية لجهود إصلاح المشكلات.

4. الخاتمة والعمل المستقبلي

وفي الختام نرى أنه في حين أن أدوات الذكاء الاصطناعي التوليدي لدعم إمكانية النفاذ في تطوير المنتجات تظهر مؤشرات واعدة إلا أن المرحلة الحالية من تطويرها تثير مخاوف بشأن موثوقيتها وفعاليتها. إن التغلب على التحديات التقنية والمنهجية تظل تشكل أموراً بالغة الأهمية لإحراز التقدم في هذا المجال ومن هذه التحديات نذكر استخدام المحتوى وإنشاء أكواد البرمجة وتوفير تغذية راجعة جيدة وتوصيات لإصلاح المشكلات وإنشاء حالات اختبار ومقاييس مناسبة والتكيف مع السياقات والمتطلبات المتنوعة. ويتطلب تحقيق هذا الهدف المزيد من البحث والتعاون بين الباحثين والمطورين والمستخدمين وخبراء إمكانية النفاذ.

إن من الضرورة بمكان أن يتم التركيز في المستقبل على تعزيز موثوقية آليات التغذية الراجعة وتحسين توصيات إصلاح المشكلات وضمان التكامل الدقيق لقيم السمات داخل أدوات الذكاء الاصطناعي لإمكانية النفاذ. ويمكن أن تعمل مؤشرات الأداء المقترحة كمعيار ذا قيمة كبرى لتقييم وتعزيز نضج هذه الأدوات في تسهيل إمكانية النفاذ في تطوير المنتجات. كما تتطلب تدابير التقييم المقترحة التحقق من صحتها من خلال مراجعة الخبراء ودراسات الحالة في العالم الحقيقي لضمان أهميتها العملية وقابليتها للتطبيق. إن أحد السبل الواعدة لتطوير مجال أدوات الذكاء الاصطناعي لإمكانية النفاذ في تطوير المنتجات هو إنشاء "مجموعة بيانات ذهبية" شاملة تتألف من عينات أكواد قابلة للنفاذ. حيث ستعمل مجموعة البيانات هذه كمعيار لتقييم فعالية أدوات الذكاء الاصطناعي التوليدي في إنتاج أكواد قابلة للنفاذ وتحديد مجالات التحسين المقترحة. كما أن اختبار قدرة أدوات الذكاء الاصطناعي التوليدي الحالية على إنشاء أكواد تلبي معايير إمكانية النفاذ ومقارنة النتائج بمجموعة البيانات الذهبية هذه يمثل فرصة مثيرة للبحث في المستقبل.

يمكن الاستفادة من مقاييس الأداء الموضحة في الدراسة الحالية لقياس أي أوجه قصور لوحظت في الكود الذي تم إنشاؤه بواسطة أدوات الذكاء الاصطناعي. ويمكن للباحثين من خلال التحليل المنهجي للكود الناتج مقابل مجموعة البيانات الذهبية تحديد أنماط أوجه القصور وتقييم مدى تقصير أدوات الذكاء الاصطناعي التوليدي الحالية في إنتاج كود قابل للنفاذ. ويمكن أن يوفر هذا التحليل رؤى قيمة في المجالات التي تتطلب المزيد من التطوير والتحسين.

يمكننا من خلال البحث والتعاون المستمرين دفع تطور أدوات الذكاء الاصطناعي قدماً مما يؤدي في النهاية إلى تعزيز مشهد رقمي أكثر شمولاً ومساواة لجميع المستخدمين.

المراجع:

- [1] S. Luan, D. Yang, C. Barnaby, K. Sen, and S. Chandra, 'Aroma: Code recommendation via structural code search', Proc. ACM Program. Lang., vol. 3, no. OOPSLA, pp. 1–28, 2019.
- [2] N. Nguyen and S. Nadi, 'An empirical evaluation of GitHub copilot's code suggestions', in Proceedings of the 19th International Conference on Mining Software Repositories, 2022, pp. 1–5.
- [3] N. M. S. Surameery and M. Y. Shakor, 'Use chat gpt to solve programming bugs', Int. J. Inf. Technol. Comput. Eng., no. 31, pp. 17–22, 2023.
- [4] A. Othman, A. Dhouib, and A. Nasser Al Jabor, 'Fostering websites accessibility: A case study on the use of the Large Language Models ChatGPT for automatic remediation', in Proceedings of the 16th International Conference on Pervasive Technologies Related to Assistive Environments, 2023, pp. 707–713.

- [5] J. Liu, C. S. Xia, Y. Wang, and L. Zhang, ‘Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation’, *Adv. Neural Inf. Process. Syst.*, vol. 36, 2024.
- [6] Y. Liu et al., ‘Refining ChatGPT-generated code: Characterizing and mitigating code quality issues’, *ACM Trans. Softw. Eng. Methodol.*, 2023.
- [7] B. Yetistiren, I. Ozsoy, and E. Tuzun, ‘Assessing the quality of GitHub copilot’s code generation’, in *Proceedings of the 18th international conference on predictive models and data analytics in software engineering*, 2022, pp. 62–71.