الفصل الأول

الدراسة النظرية

نوضّح في هذا الفصل مفهوم BPMN و LLM.

-1.1 BPMN

عمليات الأعمال (Business Process) هي سلسلة من الأنشطة أو المهام المنظمة التي يقوم بها الأفراد أو الأنظمة داخل منظمة معينة أو ضمن عدة منظمات لتحقيق هدف تنظيمي محدد أو إنتاج خدمة أو منتج معين.

نموذج (BPMN) هو معيار لنمذجة عمليات الأعمال يوفر ترميزًا بيانيًا لتحديد عمليات الأعمال ضمن مخطط إنسيابي (flowchart diagram). إن الهدف من BPMN هو دعم نمذجة عملية الأعمال لكل من المستخدمين الفنيين ومستخدمي الأعمال، من خلال توفير ترميز سهل الفهم لمستخدمي الأعمال قادر على تمثيل دلالات العمليات المعقدة.

تم تصميم BPMN ليكون مفهومًا بسهولة من قبل جميع أصحاب المصلحة في الأعمال. ويشمل ذلك محللي الأعمال الذين يقومون بإنشاء وتحسين العمليات، والمطورين الفنيين المسؤولين عن تنفيذها، ومديري الأعمال الذين يراقبونها ويديرونها. وبالتالي، تعمل BPMN كلغة مشتركة، وتسد فجوة الاتصال التي تحدث بشكل متكرر بين تصميم عملية الأعمال وتنفيذها.

(صفحة 2) [The Complete Business Process Handbook: Body of Knowledge from Process Modeling to BPM, Volume I (omg.org)](https://www.omg.org/news/whitepapers/Business_Process_Model_and_Notation.pdf)

-2.1 BPMN NOTATIONS/SHAPES

إن الهدف الرئيسي من تطوير BPMN هو إنشاء تدوين (Notation) بسيط وسهل الفهم لإنشاء نماذج عمليات الأعمال، مع توفير الدلالات والآليات الأساسية للتعامل مع التعقيد الكامن في عمليات الأعمال. حيث أن النهج المتبع للتعامل مع هذين المتطلبين المتضاربين هو تنظيم الجوانب الرسومية للتدوين في فئات محددة. يوفر هذا مجموعة صغيرة من فئات التدوين حتى يتمكن قارئ مخطط BPMN من التعرف بسهولة على الأنواع الأساسية للعناصر وفهم المخطط.

(صفحة 4)

نوضح في الفقرات التالية الفئات الرسومية لمخططات BPMN:

-1.2.1 نمذجة المهام ضمن BPMN (Tasks)

تمثل المهمة نشاطاً وحيداً يتم تنفيذه ضمن عملية الأعمال وهي العنصر الأساسي في نموذج BPMN، إذ تمثل المهمة خطوة واحدة ضمن العملية الكلية. (صفحة 5)

بعض أنواع المهام:

* مهمة مستخدم (User task):

هي مهمة أو سير عمل (workflow) نموذجية يقوم فيها شخص ما بأداء المهمة بمساعدة تطبيق برمجي.



صورة 1: كيفية تمثيل مهمة المستخدم (User task) ضمن BPMN.

* المهمة اليدوية (Manual task):

هي مهمة من المتوقع تنفيذها دون مساعدة أي محرك تنفيذ أو تطبيق.



صورة 2: كيفية تمثيل المهمة اليدوية (Manual task) ضمن BPMN.

* مهمة الخدمة (Service task):

هي مهمة تستخدم نوعًا ما من الخدمة، والتي يمكن أن تكون خدمة ويب أو تطبيقًا آليًا.



صورة 2: كيفية تمثيل مهمة الخدمة (Service task) ضمن BPMN.

-2.2.1 نمذجة التدفقات ضمن BPMN (Flows)

تُمثل التدفقات سير تسلسل الأنشطة (المهام والأحداث) ضمن العملية، نوضح في الفقرات التالية أنواع التدفقات ضمن BPMN.

* تدفق التتابع (Sequence Flow):

يُستخدم لإظهار الترتيب (التسلسل) الذي ستُنفذ به الأنشطة في مخطط العملية.

صورة 3: كيفية تمثيل تدفق التتابع (Sequence flow) ضمن BPMN.

* تدفق الرسائل (Message Flow):

يُستخدم لإظهار تدفق الرسائل بين مشاركين مختلفين في العملية (كيانات الأعمال أو أدوار الأعمال) الذين يرسلونها ويتلقونها.



صورة 3: كيفية تمثيل تدفق الرسائل (Message flow) ضمن BPMN.

-3.2.1 نمذجة الأحداث ضمن BPMN (Events)

تُشير الأحداث إلى شيء يحدث خلال سير العملية، تُستخدم لالتقاط وتمثيل الحوادث التي تؤثر على تدفق العملية، نوضح في الفقرات التالية أنواع الأحداث ضمن BPMN.

* حدث البداية (Start event):

تُشير أحداث البدء إلى بداية عملية أو عملية فرعية ولا تكون مرتبطة بتدفق تسلسل وارد. يمكن أن تحتوي العملية الرئيسية على أكثر من حدث بدء واحد، لكن العملية الفرعية تحتوي على حدث بدء واحد فقط.



صورة 4: كيفية تمثيل حدث البداية (Start event) ضمن BPMN.

* الحدث الوسيط (Intermediate event):

تشير الأحداث الوسيطة إلى شيء يحدث أو قد يحدث خلال سير العملية، بين حدث البداية وحدث النهاية.



صورة 5: كيفية تمثيل الحدث الوسيط (Intermediate event) ضمن BPMN.

* حدث النهاية (End event):

تشير أحداث النهاية إلى المكان الذي ينتهي فيه أحد المسارات في العملية. يمكن أن تحتوي العملية على أكثر من نقطة نهاية واحدة. تنتهي العملية عندما تنتهي جميع المسارات النشطة. لا تحتوي أحداث النهاية على تدفقات تسلسل صادرة.



صورة 5: كيفية تمثيل الحدث النهائي (End event) ضمن BPMN.

-4.2.1 نمذجة البوابات ضمن BPMN (Gateways)

البوابات هي عناصر تتحكم في تدفق العملية من خلال تحديد كيفية تلاقي المسارات أو تباعدها أو تقسيمها ودمجها. تساعد البوابات في اتخاذ القرارات، وإدارة المسارات المتعددة، والتحكم في تدفق الأنشطة بناءً على شروط معينة. تُعد البوابات أساسية لنمذجة منطق العملية المعقد وضمان قدرة العمليات على التعامل مع سيناريوهات مختلفة، نوضح في الفقرات التالية أنواع البوابات ضمن BPMN.

* بوابة XOR (Exclusive Gateway):

تُستخدم لتوجيه التدفق إلى واحد من عدة مسارات ممكنة بناءً على شرط. يتم اتخاذ مسار واحد فقط.



صورة 6: كيفية تمثيل بوابة XOR (Exclusive Gateway) ضمن BPMN.

* بوابة OR (Inclusive Gateway):

تُستخدم لتوجيه التدفق إلى واحد أو أكثر من عدة مسارات ممكنة بناءً على الشروط. يمكن اتخاذ مسارات متعددة في وقت واحد.

صورة 7: كيفية تمثيل بوابة OR (Inclusive Gateway) ضمن BPMN.

* بوابة AND (Parallel Gateway):

تُستخدم لتقسيم التدفق إلى عدة مسارات متوازية أو لمزامنة عدة مسارات متوازية في تدفق واحد. تُنفذ جميع المسارات في وقت واحد.



صورة 7: كيفية تمثيل بوابة AND (Parallel Gateway) ضمن BPMN.

-4.2.1 نمذجة المشاركين ضمن BPMN (Pools & Lanes)

تُستخدم أحواض السباحة (Pools) ومسارات السباحة (Lanes) لتنظيم وتصنيف المشاركين والأدوار المختلفة ضمن عملية الأعمال. تساعد هذه العناصر في توضيح الأدوار والمسؤوليات والتفاعلات بين الكيانات المختلفة المشاركة في العملية.

* أحواض السباحة (Pools):

تمثل المشاركين الرئيسيين في العملية، مثل المنظمات. كل حوض عادةً ما يمثل كيانًا أو منظمة منفصلة تشارك في العملية. تساعد المجمعات في فصل وتمييز المشاركين المختلفين في العملية بصريًا.

* مسارات السباحة (Lanes):

تُستخدم مسارات السباحة داخل الحوض لتقسيم العملية بشكل أكبر إلى أدوار أو أقسام أو مجالات وظيفية مختلفة داخل نفس المشارك. تساعد مسارات السباحة في توضيح الدور أو القسم المحدد المسؤول عن كل جزء من العملية.



صورة 8: كيفية تمثيل الأحواض والمسارات (Pools & Lanes) ضمن BPMN.

-2.1 آليات الانتباه (Attention mechanisms)

الانتباه هو آلية في التعلم الآلي والشبكات العصبونية تمكن النماذج من التركيز على أجزاء معينة من بيانات الدخل عند توليد المخرجات. حيث تسمح للنموذج بوزن أهمية المدخلات المختلفة بشكل ديناميكي، مما يعزز قدرته على التقاط العلاقات والتبعيات داخل البيانات، بغض النظر عن المسافة بينها ضمن سلسلة الدخل.

مزايا الانتباه:

* تسمح بالحساب المتوازي (Parallelization): على عكس الشبكات العصبونية المتكررة (RNN)، التي تعالج البيانات بشكل متسلسل، تسمح آليات الانتباه بالمعالجة المتوازية للدخل. وهذا يسرع بشكل كبير من الحوسبة ويجعلها أكثر كفاءة، وخاصة بالنسبة لمجموعات البيانات الكبيرة.
* التعامل مع التبعيات طويلة الأمد (Long-Range Dependencies): تسمح للنماذج بالتقاط العلاقات بين العناصر البعيدة في ضمن سلسلة الدخل بشكل أكثر فعالية من النماذج التي تعتمد فقط على الهياكل المتكررة (RNN). وهذا أمر بالغ الأهمية للمهام حيث يكون السياق من الأجزاء السابقة من التسلسل مهمًا لفهم الأجزاء اللاحقة.

-2.1 المحولات (Transformers)

هي بنية شبكة عصبونية تُستخدم في المهام التي تتضمن معالجة بيانات متسلسلة، مثل معالجة اللغة الطبيعية وفصل الكلام. تَستخدم آليات الانتباه بشكل أساسي، حيث تسمح لها بمعالجة بيانات الإدخال بالتوازي بدلاً من التتابع، على عكس الشبكات العصبية المتكررة التقليدية (RNNs).

مرجع الفقرتين

[Attention is All you Need (neurips.cc)](https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf)

-2.1 نماذج اللغة الكبير (large language models llms)

هي نماذج لغوية إحصائية تستفيد من تقنيات التعلم العميق، وخاصة هياكل المحولات، لفهم اللغة البشرية وتوليدها. وتتميز هذه النماذج بحجمها الكبير، وغالبًا ما تحتوي على عشرات إلى مئات المليارات من المعلمات (parameters)، ويتم تدريبها على كميات هائلة من بيانات النصوص من مصادر متنوعة مثل الكتب ومواقع الويب وبيانات المحادثة.

كمثال على هذه النماذج سلسلة GPT المقدمة من شركة OpenAI، PALM المقدمة من Google و LLAMA من شركة Meta.

[2402.06196 (arxiv.org)](https://arxiv.org/pdf/2402.06196)

-2.1 تحسين نماذج اللغة الكبير (Finetune llms)

في السياقات التي تكون فيها الموارد الحسابية محدودة أو حيث يكون بناء التطبيق أمرًا ضروريًا، يجب تفضيل الهندسة التوجيهية (Prompt engineering) على الضبط الدقيق (fine tuning) لنماذج اللغات الكبيرة (LLMs). على سبيل المثال، يمكن أن تتطلب عملية الضبط الدقيق موارد هائلة وتستغرق وقتًا طويلاً، كما يتضح من السعي وراء إمكانات استدعاء الأدوات في LLMs، حيث حقق تطبيق الهندسة السريعة حصريًا معدل نجاح بنسبة 100٪ دون الحاجة إلى الضبط الدقيق [1]. علاوة على ذلك، في مجال مهام هندسة البرمجيات الآلية، على الرغم من أن النماذج التي تم ضبطها بدقة قد تُظهر أداءً فائقًا مقارنة بنظيراتها المصممة بسرعة في جوانب محددة، تظل الهندسة السريعة قادرة على تحقيق نتائج تنافسية، لا سيما عند استخدام مطالبات المحادثة التي تعزز مشاركة المستخدم وتعزز فعالية النموذج [2]. بالإضافة إلى ذلك، في مجال إدارة العمليات التجارية، تسهل الهندسة السريعة النشر الفعال للنماذج المدربة مسبقًا مع التحايل على المتطلبات الأساسية للبيانات المرتبطة عادةً بالصقل الدقيق [4]. وبالتالي، فإن الهندسة السريعة مفيدة بشكل خاص في التطبيقات التي تتطلب القدرة على التكيف وتقليل تخصيص الموارد والتنفيذ السريع.

-2.1 سجل الأحداث Event log

هو ملف منظم يسجل تسلسل الأحداث المتعلقة بعملية (business process) معينة.

-2.1 خوارزمية Split Miner

هي خوارزمية آلية تعمل على إنشاء نماذج BPMN دقيقة وبسيطة إنطلاقاً من سجلات الأحداث (Event log). وهي تعالج المشكلات الشائعة في طرق اكتشاف العملية الحالية، مثل إنتاج نماذج معقدة للغاية أو نماذج لا تتناسب مع سجل الأحداث، حيث تحقق التوازن بين بساطة النموذج وملاءمته ودقته مع الحفاظ سرعة تنفيذ عالية مقارنة بالطرق الأخرى.

[Split\_Miner\_Automated\_Discovery\_of\_Accurate\_and\_Simple\_Business\_Process\_Models\_from\_Event\_Logs-libre.pdf (d1wqtxts1xzle7.cloudfront.net)](https://d1wqtxts1xzle7.cloudfront.net/69528741/Split_Miner_Automated_Discovery_of_Accurate_and_Simple_Business_Process_Models_from_Event_Logs-libre.pdf?1631705437=&response-content-disposition=inline%3B+filename%3DSplit_Miner_Automated_Discovery_of_Accur.pdf&Expires=1723451600&Signature=gTiAn07bkU~sRA-rAYydR-nEUV1Tr-4MetWxBhpKCwnI6tw0Hj4f9CQlbI4-FPK835EVzuMpN0E~P3KgNj5NrvcPtumJ34wdohLzu-WNJf3rMMVShg1TAr62-tUZoLi-PnikrvDzDVBCr5gh4gJoDPuMgmB1NHynh~3eEYn4nOuO5wGv3mwTrL2lqxjkuVyKU4AaaTc7GVUglL01Zk26US5jaWcBXn4S-UEKfQnWnMLoOf1ymfRWT7oHyXigCzq1Zfqi~0kLYWel23gFnRjjoENTZcvNAym51e4fnbzi-jAfLtr202HJmpC8gufrFeqJZ-tTElBjqNHOtytQjiNmAg__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA)

-2.1 Prompt engineering

التوجيه (Prompt) هو نص يتم تقديمه لنموذج لغوي لمساعدته على توليد استجابة.

الهندسة التوجيهية (Prompt engineering) هي عملية صياغة وتحسين التوجيهات للتواصل بشكل فعال مع نماذج اللغات الكبيرة (LLMs). هذه العملية مهمة للحصول على ردود دقيقة وذات صلة من النموذج. مع تطور نماذج اللغة، أصبحت مهارة هندسة المطالبات أساسية للمستخدمين الذين يريدون الاستفادة القصوى من نماذج اللغات الكبيرة وتحقيق أفضل النتائج في مختلف المجالات.

عند تصميم هذه التوجيهات يجب مراعاة المعايير التالية:

* الوضوح: يجب أن تكون التوجيهات واضحة وسهلة الفهم، حيث يساعد ذلك على توليد إستجابة أكثر دقة من قبل النماذج اللغوية (LLMs).
* إضافة قيود صريحة: يجب إضافة إرشادات وقيود محددة عند الطلب، حيث يساعد ذلك في تضييق نطاق تركيز النموذج اللغوي مما يؤدي إلى استجابة ذات صلة بالطلب.
* التجريب: يجب تجريب أنواع مختلفة من التوجيهات لمعرفة ما هو الأفضل، حيث أن تجربة تنسيقات مختلفة يمكن أن يساعد في اكتشاف طرق فعالة للتفاعل النموذج اللغوي.
* تحسين التوجيهات باستمرار: يجب الاستمرار في تحسين التوجيهات بناءً على النتائج التي يعيدها النموذج اللغوي، حيث أن هذه العملية التكرارية يمكن أن تعمل على تحسين التوجيه بشكل كبير بمرور الوقت.
* التحكم بمعلمات النموذج: يمكن أن يؤدي تغيير درجة الحرارة الخاصة بالنموذج (temperature) التي تحدد مدى إبداع النموذج إلى نتائج مختلفة، كما أن استخدام سلسلة من التوجيهات واحدة تلو الأخرى يؤدي إلى إنشاء تفاعلات أكثر تعقيداً مع النموذج.
* تقديم معلومات إضافية للتوجيه: إن إضافة سياق المهمة المطلوبة إلى التوجيهات يمكن أن تساعد في إنتاج استجابات أكثر دقة وملاءمة. يكون هذا مفيدًا بشكل خاص عند التعامل مع مفاهيم مجردة أو مجالات متخصصة.
* تقديم أمثلة: يمكن أن يساعد إضافة مجموعة من الأمثلة للخرج المتوقع (Few-shot learning) ضمن التوجيه على الوصول لخرج لأكثر دقة، حيث أن التعلم من خلال عدد قليل من الأمثلة يجعل النماذج قابلة للتكيف، خاصة في السيناريوهات ذات البيانات المحدودة، كما أن هذا الأسلوب يقلل من الإفراط في التجهيز(overfitting) وتعزز المرونة والتخصيص والتكيف السريع مع المهام الجديدة. [arxiv.org/pdf/2406.18678](https://arxiv.org/pdf/2406.18678)

[**f94a60a2566eade6c63a19601bcf39b4.pdf (d197for5662m48.cloudfront.net)**](https://d197for5662m48.cloudfront.net/documents/publicationstatus/174684/preprint_pdf/f94a60a2566eade6c63a19601bcf39b4.pdf)