



جامعة دمشق
كلية الهندسة المعلوماتية
قسم الذكاء الصناعي

وكيل تخاطبي باللغة العربية لإدارة طلبات الزبائن للطعام

مشروع تخرج أُعدَّ لنيل درجة الإجازة في الهندسة المعلوماتية

إعداد الطالبات

ديمة قاوق	بتول ايبو
صفاء الشايب	بشرى الشبيب

بإشراف

الدكتورة ندى غنيم
الدكتور عمار جوخدار

العام الدراسي

2017 - 2016

Damascus University

Faculty of Information Technology Engineering

Artificial Intelligence Department



Arabic Conversational Agent for Food Ordering System

**Graduation Project prepared to obtain a diploma (Engineer) of
Information Technology Engineering degree**

By

Batul Epo

Dema Qawooq

Bushra Alshbib

Safaa Alshayeb

Supervised by

Dr. Nada Ghneim

Dr. Ammar Joukhadar

Academic Year

2016 - 2017

إهداء

إلى البريق الذي أراه في عينيه كلما نظرت إليه , إلى من كنت له الأمل الذي راوده في حياته فحلم أن يراني في مثل هذا اليوم, إلى الروح التي سكنت روحي....

أبي ..

إلى من لا يمكن للكلمات أن توفي حقها, إلى من لا يمكن للأرقام أن تحصي فضائلها
إلى من حصد الأشواك ليزهر دربنا بالسعادة

أمي ..

إلى توأم روحي ورفيقة دربي .. إلى صاحبة القلب الطيب والنوايا الصادقة
إلى من رافقتني منذ أن حملنا حقائب صغيرة ومعاً سرنا الدرب خطوة بخطوة

دعاء ..

إلى من أنارتا دربي وأعانتاني بالصلوات والدعوات، إلى شمعة متقدة تنير لي طريق الأمل..

عماتي ..

إلى أزهار النرجس التي تفيض حباً وطفولةً ونقاءً وعطراً
الغاليات اللاتي مازلن يحبين على أدراج العمر الأولى

إخوتي ..

ويممة قاووق

إلى مشرق سعادتي ورونق أيامي، إلى ملاذي الأكثر أماناً، إلى ملاكي، إلى القلب الأنقى، إلى حبي الذي لا يمكن أن يشيخ، إلى مهندسة حياتي...

إلى أمي

إلى معلّمي الأول، إلى أكثر من أجد عنده سعة الصدر ولين لجانب، إلى فرحة القلب التي أرقبها دائماً عند تلقّيه لكل لحظة نجاح أحققها...

إلى أبي

إلى السند الدائم، إلى البعيدة القريبة، إلى هبتي الأثمن، إلى صديق طفولتي المشاكس، إلى الحبيب المدلّل...

إلى أخوتي منار، نور، هبة، أحمد ومحمد

إلى من جعلهم الله قربي دائماً، إلى من كانوا أول المشجّعين والمهتّئين، إلى أجمل نعم الله عليّ...

عائلتي الكريمة

إلى خير من وجّه وعلم وأعطى، إلى من مهّدوا الدّرب وأناروه...

إليكم أساتذتي

إلى التي أبت سنواتها العصيبة إلّا أن تلازم سنواتي الجامعية الخمس، إلى غصّة الفرح، إلى من أشتاق مبسمها وإشراقة السّلام في أرضها...

إلى شامي

صفاء الشايب

كطفل عيناه نحو الشمس يمضي بخطوات متعثرة ويعلو في الهواء دون أن يعرف من يصنع له تحت كل خطوة دَرَجَة، فلما أعمى عينيه لهيبُ نجاحٍ متواضع، خفضها فأبصر كل تلك الأيدي الكريمة..

إلى كل من وقف بجانبني

إلى كل من سكب في عقلي قطرةً من نور

لأن الجهود المخلصة المستمرة وإن كانت صغيرة لا شك إن تعاضدت أنها ستصنع نحو الشمس سُلماً..

أساتذتي ومدرسي

شكراً لكم

لأن السفر الطويل لا يقصُر إلا بالصحبة، فننسى طول السفر ولا نذكر إلا تلك اللحظات الجميلة التي تقاسمنا سعادتها وحزنها معاً..

أصدقائي

إلى عالمي الصغير بشمسه وقمره وكواكبه السبعة ونجومه الصغيرة الثمانية الحاملة..

عائلي

إلى الطيف الذي لا يزول والألم الذي لا ينتهي والجرح الذي لا يندمل، على أمل الشفاء باللقاء، إليك أهدي ما لم تعلم بحصوله، فقد غبت قبل أن أبدأ وأرجو أن تأتي قبل أن أنتهي..

أبي

إلى التي تحيل دمعتي نبع حب فتكون بسمتها نور ذكرى تضيء لي كلما أومضت سعادتي كالبرق في الظلماء تشاركني نهاية كل نجاح وتطلّ على الأفق وترقب لحظة الراحة التي أخذها بعد كل عناء كي تحمي هدوئي من رعونة الصخب..

أمي

بشرى سليم الشبيب

إلى من طمحا دوماً لأكون الأفضل .. وكانت دعواتهما سبب توفيقني
إلى من انتظرا هذه اللحظة بفارغ الصبر...

إلى الأب المضحّي، والأم الحنون
والداي

إلى سبب سعادتي ومصدر قوتي ... إلى الأعلى على قلبي ..
بسمة الحياة ونور الأيام، نبض الحب وجمال القلب...

بسمة، براءة، بشرى، محمدطه أخوتي

إلى من شاركوني درب الطويل .. وكانو إخوة وسنداً.. إلى من سأفتقدهم وأتمنى أن يفتقدوني...
رفقاء الدرب

إلى السند الذي رحل عني منذ بداية مشواري.. من افتقدته كثيراً وافتقده الآن أكثر...
أخي الشهيد لروحه السلام

إلى من صبر وعاش وانتظر بلهفة .. إلى نبض قلبي ونور حياتي .. إلى رفيق النصف الثاني من الطريق...
غيث

إلى كل من وقف بقربي حتى وصلت إلى هنا... إلى كل من تمنى لي الخير والنجاح ودعا لي...
شكراً من القلب...

بتول عبد العزيز ايبو

طويتم سنوات عمركم جُملاً رصينة وكتباً ثمينة وقدمتموها لنا بغير كثيرٍ تعبٍ منّا، اختزلتم الزمنَ في لحظات وقدمتم التعبَ دون انتظارِ الثمن، لكم نهدي بعضاً ممّا أهديتمونا..

الدكتورة ندى غنيم

الدكتور عمار جوخدار

ملخص:

أصبحت نظم الإجابة على الأسئلة Question Answering في الآونة الأخيرة، واحدة من أهم التطبيقات التي يتم العمل عليها في مجال معالجة اللغات الطبيعية، وبالنسبة للإجابة على الأسئلة باللغة العربية Arabic Question Answering لم تنل نفس الاهتمام عموماً، فتحديات اللغة العربية وغياب الموارد جعل من الصعب توفير نظام QA عربي ذو دقة عالية، ففي حين تكون الدقة مقبولة في بعض الأنظمة العامة إلا أن التخصص في مجال معين، يمكّن من الوصول لدقة مناسبة ويحقق الأهداف المرجوة.

في هذه الأطروحة سنقدم **Pizzanabot بيتزانابوت** وهو ليس مجرد نظام إجابة على الأسئلة وإنما هو خدمة موجهة للمطاعم وزبائنها، الهدف الأساسي منها طلب الطعام وإجراء حوار تفاعلي بين الزبون والمطعم، وبالتالي العنصر التخاطبي هنا هو الجزء الأكثر أهمية في النظام، ليمثل وكيلاً تخاطبياً يستطيع المستخدم من خلاله التخاطب مع النظام بشكل مباشر عن طريق الكتابة (Chat) أو عن طريق الصوت (حوار صوتي)، فيمكنه من الاستفسار عن شيء ما، مثل أن يسأل عن أنواع البيتزا الموجودة في المطعم، أو يسأل عن سعر بيتزا معينة، أو يطلب بيتزا، أو... إلخ، وفق محادثة تفاعلية تتم باللغة العربية وباللهجة العامية (المحكية) تنتهي بعرض الفاتورة للزبون متضمنة طلبه والسعر الموافق للطلب ويتم ذلك بعد الاتفاق على وقت ومكان وكيفية توصيل الطلب.

توجد العديد من التطبيقات والمواقع الالكترونية التي تحقق هدف طلب الطعام مثل التطبيقات الحالية التي لا تستخدم خوارزميات الذكاء الصناعي بل تعتمد على بناء قواعد معطيات ومجموعة واجهات غير تخاطبية تتيح للمستخدم اختيار طلبه من عدة خيارات متاحة أمامه كما هو الحال في تطبيق BeeOrder، الذي لاقى انتشاراً واسعاً في الآونة الأخيرة. وتجدر الإشارة إلى أن مثل هذه التطبيقات لا يمكن أن تلبي جميع حاجات الزبون، أما في الوكيل التخاطبي المقترح فسيتمكن الزبون من التفاعل مع النظام وبلهجته التي يتحدث بها دون تكلف.

وبالإضافة إلى ذلك، تأتي أهمية النظام المقترح كونه لم يسبق أن وجد نظام حوار تفاعلي عربي مشابه، ولهذا السبب تكون أبحاث كهذه تتم على اللغة العربية بشكل عام واللهجة العامية بشكل خاص تزيد من المعرفة في مجال أنظمة الحوار العربية وتشجع على العمل به مما يزيد من رصيد الأبحاث والدراسات العربية الموجهة نحو تقنية ولغويات المعلومات التي تنامت في السنوات الأخيرة.

ولقد قمنا بتطوير هذا النظام بدقة جيدة تضمن في معظم الأحيان إتمام الهدف الذي وجد من أجله، لكن لعدم وجود أنظمة مشابهة لن نستطيع أن نقول أننا تفوقنا بالدقة على مثل هكذا أنظمة عنيت باللغة العربية.

فهرس الموضوعات :

ملخص:	8
1. الفصل الأول	14
مدخل تعريفى بالمشروع	14
1.1 المقدمة:	15
2.1 هدف المشروع:	18
2. الفصل الثانى	19
الدراسة المرجعية	19
1.2 تعريف بنظم الإجابة على الأسئلة:	20
1.1.2 نظم الإجابة على الأسئلة Question Answering System:	20
2.1.2 البنيان العام لنظم الإجابة على الأسئلة:	20
3.1.2 نظرة تاريخية لنظم الإجابة على الأسئلة:	20
2.2 الدراسة المرجعية:	22
1.2.2 نظم الإجابة على الأسئلة باللغة العربية:	22
2.2.2 نظرة تاريخية إلى التعرف على الكلام Automatic Speech Recognition:	34
3.2.2 لمحة عن منهجيات الاختبار المتبعة في نظم الإجابة عن الأسئلة والوكلاء التخاطبية:	38
3. الفصل الثالث	41
التحليل	41
1.3 المتطلبات الوظيفية:	42
2.3 المتطلبات غير الوظيفية:	42
3.3 فئات المستخدمين المستهدفين:	43

43	4.3 حالات استخدام النظام:
47	4. الفصل الرابع
47	التصميم البنيوي
48	1. مقدمة:
48	2. طريقة العمل المقترحة لتحقيق نظام حوار بالغة العربية العامية:
50	3. جمع وتحليل البيانات Dialog corpus:
50	1.3 جمع البيانات:
50	2.3 تصنيف الكلمات المفتاحية :
51	4. نظام التعرف على الصوت:
53	5. عمليات قبل المعالجة (preprocessing):
53	1.5 التقطيع Tokenizing:
53	2.5 إصلاح الدخل Repairing:
53	6. استخراج الكلمات المفتاحية keywords extraction:
54	1.6 مقدمة عن LingPipe API :
54	2.6 كشف أسماء الكيانات Named entity recognition (NER):
55	1.2.6 كشف أسماء الكيانات في LingPipe:
57	3.6 التجذيع Stemming:
57	7. تحليل نمط الدخل Intention analysis:
58	1.7 أنماط جمل الزبائن intention types:
58	2.7 التحليل اللغوي Syntax analysis :
61	8. الاستجابة التفاعلية interactive response:

61	1.8 إجابة من نمط S-What
61	2.8 إجابة من نمط S-Have
62	3.8 إجابة من نمط S-Price
62	4.8 إجابة من نمط S-WAnt
63	الأنطولوجيا (Ontology):
63	1. تعريف الأنطولوجيا (Ontology):
65	2. مكونات الأنطولوجيا (Ontology):
66	3. أهداف الأنطولوجيا (Ontology):
66	4. أدوات بناء الأنطولوجيا (Ontology):
67	5. الأنطولوجيا في أنظمة الحوار:
77	5. الفصل الخامس
77	التصميم التفصيلي والتحقيق البرمجي
78	1. وصف بيئة التنفيذ:
78	2. تحقيق طبقة ال Web API:
79	1.2 أهم توابع ال FoodOredring Web API:
79	2.2 أهم الصفوف المستخدمة لتحقيق FoodOredring Web API:
82	3. بناء نظام طلبات مبسّط اعتماداً على ال FoodOrdering Web API:
83	4. أهم المكتبات البرمجية والأدوات والمحركات المستخدمة:
84	5. كيفية تحقيق المتطلبات غير الوظيفية برمجياً:
85	6. الفصل السادس
85	التنفيذ والاختبارات

1. نتائج التنفيذ:	86
1.1. إضافة صاحب المطعم لأطباقه ومكونات كل طبق لتوسعة الأنتولوجيا بها:	86
2.1. المحادثة بين الزبون والوكيل التخاطبي:	88
2. المنهجية المقترحة لاختبار النظام ونتائجها:	92
1.2. اختبار مرحلة تحويل الصوت لنص مكتوب:	92
2.2. اختبار مرحلة المعالجة النصية:	93
تحليل نتائج الاختبارات:	96
آفاق وأعمال مستقبلية:	97
المراجع:	98

1. الفصل الأول

مدخل تعريفي بالمشروع

1.1 المقدمة:

تُعرّف المعالجة الآلية بالجهود التي تحاول إزالة الحواجز ما بين اللغة التي يستخدمها الإنسان العادي في ظروفه الطبيعية، والحاسب كآلة ذات قدرات عالية في فهرسة وتخزين ومعالجة واستدعاء البيانات والمعلومات، بما يجعل الإنسان قادر على استثمار أقصى طاقات وإمكانات الحاسب بسهولة ويسر وعبر لغة التعامل الطبيعية، وبما يجعل الحاسب قادر على أن يفهم لغة الإنسان الطبيعية على مستوى الكلمة والجملة والمعنى وينفذ ما يريده الإنسان عبر هذا الفهم، وإن فهم الحاسب للغات الطبيعية من المشكلات الكبرى التي تواجه نظم المعالجة الآلية، فلذلك جرت أبحاث عديدة في ذلك المجال واستطاعت المعالجة الآلية عبر رحلتها التاريخية وما أنجزته خلالها من نجاحات وتطورات متنوعة، أن تفتح لنفسها مجالات للتطبيق العملي في مسارات متنوعة، بعضها بدأنا نلمس آثاره عملياً مثل أدوات الترجمة الآلية المتوفرة حالياً مجاناً على محركات البحث الشهيرة، وبعضها الآخر لا يزال في البدايات ولم يحقق الانتشار والشعبية المطلوبة، ومن تطبيقات المعالجة الآلية للغة: الترجمة الآلية، التلخيص الآلي، التوليد الآلي للغة، استخلاص المعلومات، استرجاع المعلومات، الإجابة على الأسئلة، التنقيب في النصوص، تحويل النص إلى كلام منطوق وفهم الصوت. وتقسم تلك التطبيقات إلى صنفين رئيسيين، تطبيقات معتمدة على النص text-based وتطبيقات معتمدة على الحوار dialog-based، النوع الأول يتضمن معالجة النصوص المكتوبة، والأبحاث التي تندرج تحت هذا النوع أصبحت مستخدمة في تطبيقات استخراج المعلومات من الرسائل أو المقالات، الترجمة اللغوية، وتلخيص النصوص، وغيرها. أما بالنسبة للتطبيقات من النوع الثاني فهي تشمل التواصل بين الآلة والإنسان باللغة الطبيعية، وأهمها الإجابة على الأسئلة، وقد نالت، هذه الأخيرة، نصيباً عالياً من اهتمام الباحثين، حيث تفترض أن الحاسب يستطيع الإجابة على أي أسئلة باللغة الطبيعية من خلال البحث في مجموعة ضخمة من المستندات والوثائق أو في قاعدة معطيات، وفي حال أصبحت نظم الإجابة على الأسئلة تفاعلية بين المستخدم والحاسب، هنا نأتي لما يسمى نظم الحوار، والتي ظهرت كجزء من نظم الإجابة على الأسئلة بهدف تقديم خدمات للزبائن والإجابة عن تساؤلاتهم وتحقيق رغباتهم وفق حوار تفاعلي أشبه بالحوار بين شخصين، وفي عالم التقنية لوحظ موجة جديدة تحت مجال أنظمة الحوار بدأت بالانتشار سريعاً خلال الشهور القليلة الماضية وهي موجة الـ Chatbots أو روبوتات الدردشة، وتمثل وكيل تخاطبي Conversational agent وهو برنامج حاسوبي يحاول محاكاة المحادثات بين الآلة والإنسان،

وتقوم Chatbots بمهام معينة وكثيرة تبدأ من حجز غرفة في فندق وتنتهي بقول "نكتة" وكل ذلك حسب طلب المستخدم ورغبته.

مبررات العمل :

بدأت معظم الشركات الكبرى بدعم روبوتات الدردشة في تطبيقاتها، مثل فيس بوك وجوجل وتيلغرام وسكايب وغيرها، وهي ليست بالأمر الجديد، حيث ظهرت قبل سنوات طويلة على الإنترنت كما دعمتها بعض تطبيقات المحادثة القديمة، لكنها كانت غير ذات فائدة تقريباً إذ كانت مجرد تجارب بإمكانيات محدودة جداً، ثم تطورت مع تطور تقنيات الذكاء الاصطناعي Artificial Intelligence وتعلم الآلة Machine Learning، إضافة إلى تقنيات معالجة اللغة الطبيعية NLP، وأصبحت البرامج أكثر قدرة على فهم ما يكتبه الإنسان، وعلى تقديم المساعدة له.

ويُعزى اهتمام الشركات الكبرى بروبوتات الدردشة لعدة أسباب، الأول هو تطور تقنيات الذكاء الاصطناعي - كما ذكرنا - التي باتت قادرة على فهم المستخدم بشكل أكبر، وثانياً لأننا بدأنا نتحول شيئاً فشيئاً من عصر طلب المساعدة عبر الويب أو عبر تطبيقات الهاتف الذكي، إلى عصر المُساعد الشخصي Personal Assistant الذي يفهم ما يريده الشخص منه بالضبط، ويقدم له المساعدة بشكل فوري، وهذا مكانٌ بدأت تجد الشركات فيه مصدراً للربح أيضاً، حيث سنتمكن جميعاً خلال فترة قريبة من طلب الطعام وسيارات الأجرة والتسوق من خلال الدردشة.

وقد انتشرت حديثاً - في محيطنا - بشكل كبير التطبيقات المتعلقة بالمطاعم وطلب الطعام، فأصبحنا نسمع كثيراً عن اشتراك عدة مطاعم بتطبيق معين، يؤمن للمستخدم طلب ما يريده عن طريق البحث بين قوائم التطبيق وتحديد الوجبة المطلوبة، وبذلك تقتصر هكذا تطبيقات لعنصر التفاعل والحوار بين النظام والزيون ، ولدمج التطور التقني الذي يظهر بانتشار Chatbots والتوجه المتسارع نحو تطبيقات طلب الطعام، اخترنا أن يكون البحث عبارة عن وكيل تخاطبي لإدارة طلبات الزبائن للطعام.

ولا بد في نظم الحوار من تحديد لغة التواصل بين الإنسان والآلة، وقد جرت الأبحاث على العديد من اللغات وظهرت الكثير من تطبيقات نظم الحوار الذكية بتلك اللغات، ولكن لم تتل اللغة العربية النصيب نفسه من الدراسات والاهتمامات نظراً لقلّة الموارد العربية وصعوبة تلك اللغة، فتوجهنا لاختيار اللغة العربية

كلغة تخاطب مما يزيد من رصيد الأبحاث والدراسات العربية الموجهة نحو تقنية ولغويات المعلومات التي تنامت في السنوات الأخيرة ويشجع على العمل بها.

وبسبب كون التطبيق تفاعلياً موجهاً لجميع الناس من جميع الأصناف والأعمار، وكونه متخصصاً بطلب الطعام، ارتأينا أن يكون التفاعل باللهجة العامية، فأسهل للزبون "في مجتمعنا" أن يقول "بدي وحدة بيتزا فصول أربعة" من أن يقول "أريد قطعة واحدة من بيتزا الفصول الأربعة"، فاختيار اللهجة العامية يزيد من أهمية بحثنا، ولصعوبة وجود قواعد تحكم الطريقة التي يتكلم بها الناس باللهجة العامية تم تخصيص التطبيق في البداية ليكون موجهاً لمطاعم البيتزا.

جميع هذه الأسباب قادت بنا لاختيار هذا البحث، ودراسة كيفية إمكانية تحقيق وكيل تخاطبي باللهجة العامية العربية يقدم خدمة للمطاعم في إدارة طلبات الزبائن، ويساعد الزبائن في طلب الطعام.

2.1 هدف المشروع:

بعد ازدياد التوجه الكبير للمطاعم في الآونة الأخيرة نحو تطبيقات طلب الطعام، والتي أثبتت كفاءتها في زيادة الأرباح، فوفقاً لدراسات عديدة، مثل هذه التطبيقات تزيد نسبة الطلب 23%، كان هدف المشروع منذ بدايته إلى جانب زيادة رصيد الأبحاث في لغتنا العربية، الوصول لتطبيق يتفوق على تطبيقات طلب الطعام المنتشرة ويكون فكرة سباق في الحصول على أول Chatbot عربي عامي مختص بطلب الطعام، وتحديدًا البييتزا.

وقد صُمم التطبيق ليقدم الخدمات الأساسية المتعلقة بطلب الطعام، فيستطيع الزبون من خلاله تبادل الحوار باللهجة العامية مع النظام إما صوتياً أو كتابياً عبر واجهات سهلة للاستخدام، مما يحقق وصول التطبيق لأكبر شريحة من المستخدمين، فيستطيع الزبون من خلال الحوار تسجيل طلبه واستفساراته المتعلقة بالطعام، ويمكن أيضاً أن يطلب توصيل الطعام، عن طريق تحديد مكانه وينتهي الحوار بتقديم الفاتورة له.

وكغيره من الأنظمة الحوارية يجب فيه مراعاة عامل الزمن، فيكون زمن الاستجابة متناسباً مع فترة المعالجة دون أن يسبب ملل للزبون، بالإضافة لكونه لا يتبع أسلوباً موحداً في الإجابة بل يحاكي طريقة تحاور الأشخاص فيما بينهم، أما بالنسبة لتعزيز ثقة الزبون بالنظام، نحقق ذلك عن طريق زيادة دقة التعرف، ففي نهاية الحوار يقوم النظام بتأكيد الطلب للزبون وينتظر منه الموافقة ليسجل الطلب، ومن جهة أخرى يكون النظام قابلاً للتوسع لجميع أصناف الطعام ومن ثم التوسع اللهجات الأخرى، فنصل لهدفنا في نشر التطبيق الذي سيلاقي اهتماماً واسعاً من قبل الزبائن دون التطبيقات التقليدية، ويحقق الدقة المرجوة.

فلو كان أحدهم جائعاً وأراد طلب بعض الطعام، لماذا يفتح تطبيقاً خاصاً بذلك ويبحث في القوائم بحثاً عن الوجبة المناسبة، في الوقت الذي يستطيع فيه كتابة جملة واحدة لروبوت طلب الطعام مثل: "أطلب لي الدجاج بالأرز وأرسله إلى منزلي"، وبمجرد الضغط على Enter سيتم المطلوب دون بذل أي جهد آخر. وهذا هو المستقبل الذي تسعى الشركات للوصول إليه.

2. الفصل الثاني

الدراسة المرجعية

1.2 تعريف بنظم الإجابة على الأسئلة:

1.1.2 نظم الإجابة على الأسئلة QUESTION ANSWERING SYSTEM:

يعرف نظام الإجابة على الأسئلة بأنه نظام يجيب على الأسئلة التي يطرحها المستخدم باللغة الطبيعية باستخدام عدة أدوات و مصادر لغوية تهدف إلى استخراج الجواب الدقيق الموجود على الويب أو ضمن وثائق محلية.

2.1.2 البنين العام لنظم الإجابة على الأسئلة:

العديد من نظم الإجابة على الأسئلة (Question Answering System) تتشابه في مكوناتها العامة، وإن كانت تختلف أحياناً في مستوى التعقيد التصميمي الذي يبلغه كل مكون، إن المكونات الأساسية هي:

محل السؤال: دخل هذه المكونة هو السؤال، يقوم هذا الجزء بمعالجة السؤال وتحليله للحصول على المعلومات المفيدة التي يتضمنها هذا السؤال كنمط السؤال ، ونمط الجواب المتوقع.

محرك البحث: الذي يستخرج الملفات التي يمكن أن تحتوي الجواب الصحيح.

محل الوثائق: المستردة من قبل محرك البحث، يقوم باستخراج الفقرات التي يمكن أن تحتوي على الأجوبة، وتختلف التقنية المستخدمة في هذه المكونة، فهناك أساليب التحليل النحوي، والدلالي، وبعض الطرق الإحصائية .

مركب الجواب: الذي يقوم بترتيب الفقرات المنتقاة من قبل محل الوثائق اعتماداً على معايير كثيرة ثم يقدم للمستخدم الأجوبة النهائية .

تمثل هذه المكونات السابقة البنين العام لنظم الإجابة على الأسئلة، ليس بالضرورة أن تلتزم جميع النظم بهذه البنية، فالكثير من النظم تقوم ببعض التعديلات على هذه البنية بما يلائم الخوارزميات والتقنيات التي تستخدمها.

3.1.2 نظرة تاريخية لنظم الإجابة على الأسئلة:

إن بدايات هذا النوع من النظم كانت مع النظام Baseball الذي تم تنجيزه في عام 1961 حيث كان لديه قدرة الإجابة على الأسئلة الموجهة بلغة طبيعية محدودة جداً والتي كانت تتعلق بمعلومات مخزنة مسبقاً في

قاعدة معطيات محلية تحوي معلومات عن اتحاد البيسبول في أمريكا US baseball league يقوم النظام بتحليل السؤال نحويًا، يستخرج المعلومات المهمة منه، ثم يحول هذه المعلومات إلى صيغة مناسبة لطريقة البحث المتبعة من قبل النظام.

يقوم النظام Lunar المطور من قبل (WALTZ, 1975) بالإجابة على الأسئلة بالطريقة نفسها المتبعة من قبل نظام Baseball، ولكن يتم البحث ضمن مجموعة أكثر تعقيداً من الملفات النصية تحوي معلومات عن التحليل الجيولوجي للصخور.

في عام 1977 قام (Hendrix, 1977) بتطوير النظام LIFER الذي يؤمن واجهة تخاطب بلغة طبيعية بين المستخدمين ونظم قواعد المعطيات، حيث كان لديه قدرة كبيرة على التحليل النحوي وبعض إمكانيات التحليل الدلالي.

في عام 1982 طور (Wilensky R., 1984) نظام استعلام لمساعدة مستخدمي نظام UNIX سمي مستشار يونيكس Unix Consultant UC، حيث يوفر هذا النظام واجهة للتخاطب مع مستخدمي نظام UNIX المتحدثين باللغة الإنكليزية يقوم بالإجابة على أسئلتهم المطروحة بلغتهم الطبيعية والمتعلقة بتعليمات وطرق استخدام نظام التشغيل UNIX (هناك واجهات للعديد من اللغات الأخرى غير الإنكليزية).

تعتمد جميع النظم المشار إليها سابقاً على الإجابة ضمن مجال محدد restricted-domain QA system وقد توصلت لنتائج جيدة ضمن المجال الذي حددها.

يعتبر النظام Start الموصف بشكل كامل من قبل (2003, Katz, Lin) أول نظام يجيب على الأسئلة ضمن أي مجال Open-domain QA System، بدأ العمل به منذ العام 1988 وأصبح متاحاً على شبكة الإنترنت منذ عام 1993 ليجيب على ملايين الأسئلة المطروحة من قبل آلاف المستخدمين في العالم، حيث يجيب النظام على مختلف أنواع الأسئلة كالأسئلة المنطقية، وأسئلة الحقائق، والأسئلة المعقدة منها والبسيطة، وبكثير من اللغات.

2.2 الدراسة المرجعية:

1.2.2 نظم الإجابة على الأسئلة باللغة العربية:

تطورت نظم الإجابة على الأسئلة منذ نشأتها عام 1960م، من نظم بسيطة تتعامل مع معطيات مهيكلة في قواعد معطيات أو قواعد معرفة إلى نظم متطورة طُرحت على الويب وتعالج وثائق متنوعة وغير مهيكلة، ومن نظم تقتصر على الإجابة على أسئلة الحقائق فقط إلى نظم تعالج العديد من أنواع الأسئلة ومنها الأسئلة المعقدة، في هذا الفصل نستعرض دراسة مرجعية شاملة عن أشهر نظم الإجابة على الأسئلة سواءً النظم التي تعالج اللغة العربية أو التي تعالج اللغات الأخرى .

كانت أولى محاولات بناء نظام للإجابة على الأسئلة باللغة العربية مع نظام AQAS الموصف من قبل (Mohammed et al.. 1993) الذي يقتصر على إيجاد الجواب من معطيات مهيكلة في قواعد معرفة.

ظهر بعد ذلك عدة أنظمة في محاولة للحاق بركب التطور في مجال نظم الإجابة على الأسئلة.

1.1.2.2 : نظام QARAB

الهدف الأساسي من نظام QARAB (Hammo.b et al., 2002) هو تحديد الفقرات التي تجيب على السؤال المطروح باللغة العربية ، فيأخذ الأسئلة المطروحة ويحاول تقديم إجابات قصيرة. المصدر الأساسي للمعرفة هو مجموعة من النصوص الصحفية العربية المستخرجة من الراية، وهي صحيفة نشرت في قطر، يستعمل طريقة مبسطة لمعالجة السؤال ،ولا يصل إلى مستوى دلالي لفهم السؤال.

و لضمان الإجابة على السؤال يفترض النظام مايلي :

- الجواب موجود في مجموعة الوثائق التي يستخدمها النظام للإجابة عن الأسئلة (مقالات جريدة الراية)
- لايمتد الجواب على السؤال الواحد لأكثر من وثيقة
- الجواب هو فقرة قصيرة .

بنية نظام QARAB:

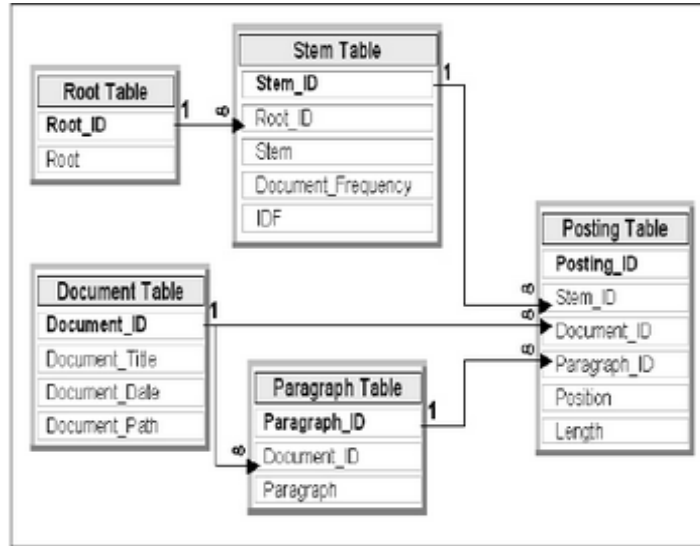
يستخدم نظام QARAB مكونين رئيسيين :

○ نظام استرجاع المعلومات:

الهدف الرئيسي منه إيجاد مجموعة الوثائق المتعلقة بالاستعلام الممرر و المستخرج من السؤال المطروح. تم بناء هذا النظام من الصفر وذلك باستخدام نموذج الفضاء الشعاعي كما تم استخدام نظام إدارة قواعد المعطيات العلاقاتية (RDBMS) المقترح من قبل [Lundquist et al., 1999]. أما بالنسبة للتجذير فقد تم استخدام خوارزمية خوجة (khoja, 1999).

بالنسبة لمخطط قواعد المعطيات (الشكل "1") الذي تم بناؤه فهو يحوي الجداول التالية :

- جدول الجذور ROOT_TABLE: لتخزين جذور الكلمات المستخرجة من وثائق جريدة الراية.
- جدول الجذوع STEM_TABLE: لتخزين جذوع الكلمات بالإضافة لتردد كل جذع في مجموعة الوثائق Frequency Document والتردد العكسي (IDF , Inverse Document Frequency).
- جدول التسجيل POSTING_TABLE: لتخزين كل حالات ورود الجذوع في وثيقة معينة من الوثائق.
- جدول الوثائق DOCUMENT_TABLE: لتخزين معلومات عن الوثيقة كالعنوان والتاريخ ومسار التخزين على القرص.
- جدول الفقرات PARAGRAPH_TABLE: لتخزين جميع الفقرات المستخرجة من وثيقة معطاة.
- جدول الاستعلام QUERY_TABLE: لتخزين معلومات عن الاستعلام، حيث يحوي كلمات الاستعلام الأصلية بالإضافة للكلمات الناتجة عن توسيع الاستعلام.

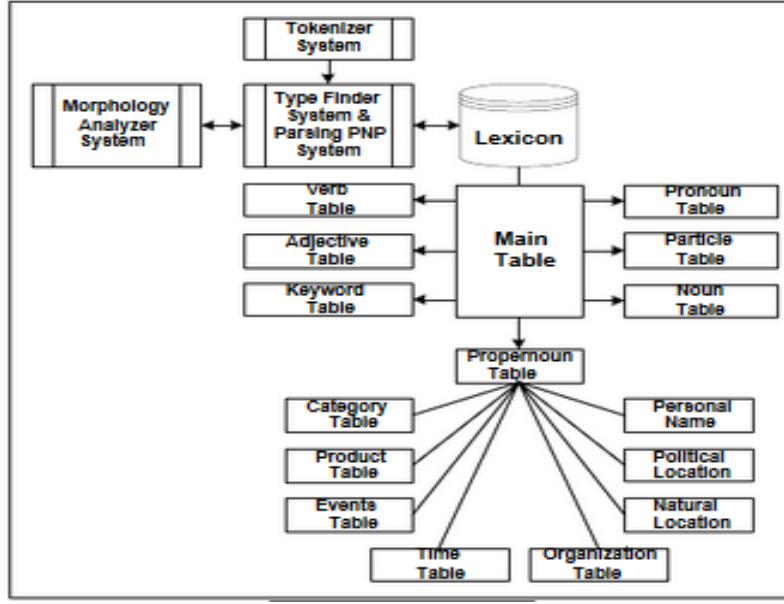


الشكل (1) مخطط قواعد المعطيات العلائقي لنظام استرجاع المعلومات.

○ نظام معالجة اللغات الطبيعية:

ويتألف من عدة مكونات :

- المقطع Tokenizer : الذي يقوم بتقطيع النص إلى مجموعة من المفردات المنفصلة Tokens.
- المكونة الخاصة بإيجاد نوع المقطع Type Finder: التي تقوم بإيجاد (POS) لكل مفردة .
- المكونة الخاصة بإيجاد بعض الخصائص Feature Finder: (مذكر أم مؤنث، مفرد أم جمع، الزمن ماض أم مضارع ..) لكل مفردة.
- المكونة الخاصة بإيجاد أسماء العلم The Proper Noun Phrase Parse: (أسماء الأشخاص والمنظمات والأمكنة والأعداد والتواريخ).



الشكل (2) البنية للمكونة الخاصة بمعالجة اللغات الطبيعية

خطوات بناء النظام:

○ معالجة السؤال في نظام QARAB:

يتعامل النظام مع السؤال الوارد كمجموعة من الكلمات "bag of words" تستخدم كاستعلام للبحث ضمن الوثائق لاسترجاع الوثائق الأكثر احتمالاً لورود الجواب فيها.

تتم عملية معالجة السؤال وتشكيل الاستعلام بعدة خطوات هي كالتالي:

- بداية نستخدم المكونة الخاصة بتقطيع النص Tokenizer لتقطيع النص إلى مفردات منفصلة.
- بعد ذلك يتم إزالة كلمات الوقف stop words.
- أما الكلمات المتبقية فيتم إيجاد أماراتها Part Of Speech وذلك لاستخراج الكلمات الهامة التي من المفترض أن ترد في الجواب، كما يتم تحديد اسماء العلم الواردة في السؤال.
- بينما لتشكيل الاستعلام يتم أخذ الكلمات المفتاحية وتشمل أفعال تتبع قوالب صرفية (مورفولوجية) معينة، وخاصة أسماء العلم والتي تعتبر الوسيلة الأفضل لإيجاد الجواب.
- حيث جرى بناء قائمة من الكلمات المفتاحية للغة العربية التي من الممكن أن تحدد أسماء كيانات (كأسماء الأشخاص والمنظمات والأماكن والأعداد والتواريخ).

- وأخيراً يتم تحديد نوع الجواب المتوقع بحسب كلمة السؤال فمثلاً السؤال الذي يبدأ بكلمة أين يحدد جواب من نوع "مكان" وفي جدول 1 قائمة بكلمات السؤال التي يعالجها النظام مع نوع الجواب المتوقع لكل منها.

Query Starting with	Query Type
من	Who, Whose Person
متى	When Date, Time
ما، ماذا	What, Which Organization, Product, Event
أين	Where Location (natural, political)
كم	How Much, How Many Number, Quantity

جدول (1) أنماط الأسئلة التي يعالجها نظام QARAB

○ معالجة الجواب في نظام QARAB :

إن الدخل لمكونة توليد الجواب هو السؤال المطروح باللغة الطبيعية ومجموعة الوثائق المسترجعة من نظام استرجاع المعلومات، بعدها نستخرج من هذه الوثائق الفقرات ذات العلاقة أو الصلة، وأخيراً لتحديد الفقرة التي تحتوي على الجواب نقوم باختيار الفقرة التي تحتوي على أكبر عدد من كلمات الاستعلام Bag of words بالإضافة إلى أسماء العلم المتوقع إيجادها في الجواب.

الاختبارات والنتائج:

بالنسبة لوثائق الاختبار تم اعتماد مجموعة وثائق من جريدة الراية القطرية .
بينما تمت عملية تقييم النظام بشكل مباشر بواسطة أربعة متحدثين أصليين باللغة العربية (خريجين جامعيين) حيث قدموا 113 سؤال للنظام وحكموا بأنفسهم على صحة الجواب.
من حيث النتائج ذكروا بأنهم حصلوا على نسبة 97.3% في كل من الدقة Precision والتغطية Recall.
ذكر (Benajiba.Y et al., 2007) بأن مثل هذه النتائج لم يتوصل إليها أي نظام من قبل لا باللغة العربية ولا حتى بأية لغة أخرى، لذا لا يمكن الوثوق بدقة هذه النتائج.

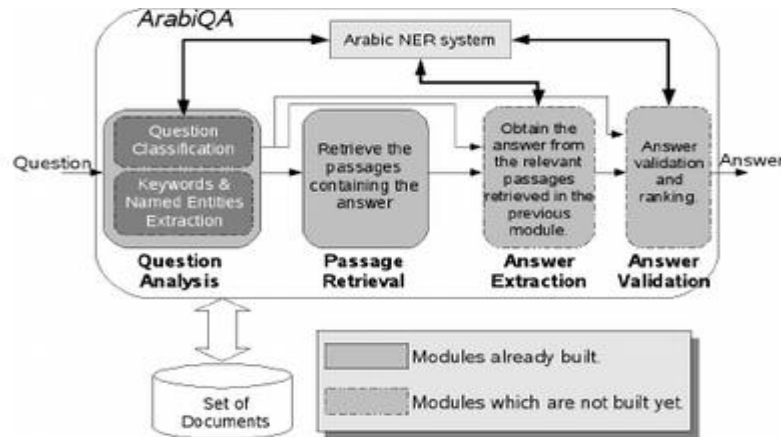
2.1.2.2 نظام ARABIQA :

نظام ARABIQA (Benajiba.Y et al 2007) للإجابة الآلية على الأسئلة يعتمد على نظام لاستخلاص المقاطع (Passage Retrieval) ونظام التعرف الآلي على أسماء العلم اسمه ANERSys، ويحتوي أيضاً على نظام لاستخلاص الجواب AE Answer Extraction من النصوص خاص بالأسئلة التقليدية (من، متى، أين ...).

البنية العامة:

تتألف البنية العامة لهذا النظام من أربع مكونات رئيسية:

- المكونة الخاصة بتحليل لسؤال: وتحدد هذه المكونة نوع السؤال (الذي يستخدم في استخراج الجواب كنوع الجواب المتوقع)، الكلمات المفتاحية (التي تشكل الاستعلام في مكونة استرجاع المعلومات)، كما يتم تحديد أسماء الكيانات التي تظهر في السؤال (والتي تستخدم في تحديد مدى صحة الجواب المقترح).
- المكونة الخاصة باسترجاع الفقرات: وتمثل هذه المكونة نواة النظام لأن فشلها في استرجاع الفقرات التي تحوي على الجواب يعني فشل النظام بشكل كامل بإيجاد الجواب.
- المكونة الخاصة باسترجاع الجواب: وتقوم باستخراج عدد من الفقرات التي تحتوي على الجواب.
- المكونة الخاصة بتحديد صحة الجواب: تحدد من أجل كل جواب احتمالية صحة هذا الجواب وترتب الأجوبة حسب احتمالية صحتها من الأجوبة الأكثر صحة إلى الأقل صحة.



الشكل (3) مخطط عمل نظام ARABIQA

نظام كشف أسماء الكيانات (ANERSYS):

يعتمد النظام Arabic Name Entity Recognition System بشكل أساسي على هذا النظام، حيث نعاني في معالجة اللغة العربية من مشكلة غياب الحروف الكبيرة في أسماء العلم مما يجعل أمر تمييز هذه الأسماء في النص أمراً صعباً. لذا تم دراسة المنهجيات المتبعة في تطوير نظم تحديد أسماء الكيانات بشكل مستقل عن اللغة Independent Language NER. وبعد هذه الدراسة وجد أن أفضل منهجية يمكن استعمالها في منهجية الإنتروبي الأعلى Maximum Entropy، لذا تم اعتماد هذه المنهجية لتدريب النظام وذلك بالاعتماد على وثائق منمطة corpora خاصة لهذا الغرض تم بناؤها بشكل يدوي وذلك لعدم توفر مثل هذه corpora للغة العربية، وهي كما ذكر Benajiba مؤلفة من 150000 مقطع منمط بالاعتماد على الصفوف المعرفة في مؤتمر CONLL 2002 هذه الصفوف هي كما يلي: أسماء الأشخاص PER، أسماء الأماكن LOC، أسماء الكيانات التي لا تنتمي للصفوف السابقة MISC، كما تم اعتماد طريقة للتنميط IOB2 التي تقسم كل صنف لقسمين (Beginning-Inside) وتضيف نمط جديد هو (Other). فيصبح مجموع الأنماط التي تم اعتمادها في عملية التنميط في corpora المذكورة أنفاً كما يلي:

Beginning-PERSON B-PERS: بداية اسم شخص.

Inside-PERSON I-PERS: تتمة اسم شخص.

Beginning-LOCation B-LOC: بداية اسم مكان.

Inside- LOCation I- LOC: تتمة اسم مكان.

Beginning-LOCation B-ORG: بداية اسم منظمة.

Inside- LOCation I- ORG: تتمة اسم منظمة.

Beginning- MISCellaneous B-MISC: بداية اسم كيان لا ينتمي إلى أي من الكيانات السابقة.

Inside- MISCellaneous I- MISC: تتمة اسم كيان لا ينتمي إلى أي من الكيانات السابقة.

Other O: الكلمة ليست اسم كيان.

لدى هذا النظام إصدارين:

• ANERsys 1.0: تعتمد على منهجية الإنتروبية العظمى (ME) Maximum

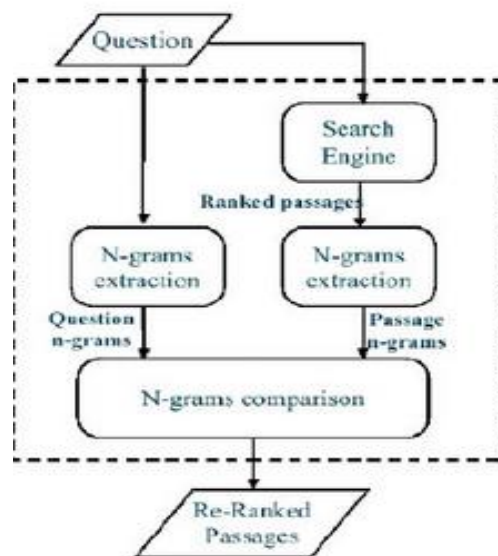
Entropy.

• ANERsys 2.0: تعتمد على منهجية الإنتروبية العظمى Maximum Entropy مع

.Part of Speech tagger

تعديل نظام استرجاع الفقرات JIRS مع اللغة العربية:

نظام JIRS هو نظام استرجاع فقرات متاح مجاناً على الانترنت، يعتمد من حيث فهرسة الوثائق على نموذج N-gram أما من حيث استرجاع الفقرات ذات العلاقة بالسؤال فإنه يستخدم لذلك نموذج كثافة المسافة Distance Density الذي يعطي أوزان أكثر للفقرات الأقرب لهيكلية السؤال، أما من أجل التجدير فقد جرى استخدام خوارزمية التجدير المخفف Light Stemming. تم تعديل هذا النظام ليلائم العمل على النصوص العربية.



الشكل (4) البنية العامة لنظام JIRS

المكونة الخاصة بتحليل السؤال:

تتم عملية تحليل السؤال وفق الخطوات التالية:

- استخراج الكلمات المفتاحية وذلك بإزالة كلمات الوقف Stop Words الواردة في السؤال.
- استخراج أسماء الكيانات التي تظهر في السؤال وذلك باستخدام نظام تحديد أسماء الكيانات .ANERsys
- إيجاد نوع السؤال وذلك تبعاً لتصنيف محدد.

تم تحديد هذا التصنيف تبعاً لإرشادات مؤتمر CLEF وهو خاص بأسئلة الحقائق فقط

○ الأسماء: أسماء الأشخاص، العناوين، الأماكن(البلدان،المدن).

○ التعاريف.

○ التواريخ: يوم، شهر، أسبوع.

○ الكميات: عملات، أبعاد، أعمار.

المكونة الخاصة باستخراج الجواب:

تم بناء هذه المكونة لأسئلة الحقائق فقط، لأن الإجابة على الأسئلة المعقدة يحتاج إلى موارد فعالة للمعالجة الدلالية والنحوية للغة العربية.

المنهجية المقترحة في هذا النظام تعتمد على خطوتين أساسيتين:

أولاً: نظام تحديد الكيانات NER يقوم بتنميط كل الكيانات في الفقرات المسترجعة من قبل نظام استرجاع الفقرات.

ثانياً: يقوم النظام بإزالة جميع أسماء الكيانات التي لا تتطابق مع نوع الجواب المتوقع مسبقاً (في عملية تحليل السؤال).

ثالثاً: وأخيراً يقدم النظام مجموعة من الأجوبة الممكنة باستخدام مجموعة من قوالب للأجوبة محددة مسبقاً.

الاختبارات والنتائج:

نظراً لعدم دعم مؤتمرات التقييم (CLEF, TREC) للغة العربية ، تم إنشاء مجموعة اختبار للغة العربية Test-bed بشكل يدوي وذلك من وثائق الويكيبيديا العربية.

تتضمن مجموعة الاختبار:

الوثائق: حيث تم استخدام مجموعة من الوثائق تتضمن مقالات من الويكيبيديا العربية وصل تعداد هذه الوثائق إلى نحو 638.11 وثيقة (متاحة مجاناً).

الأسئلة: حيث تم وضع حوالي مئتين سؤال تبعاً لإرشادات مؤتمر CLEF 2006.

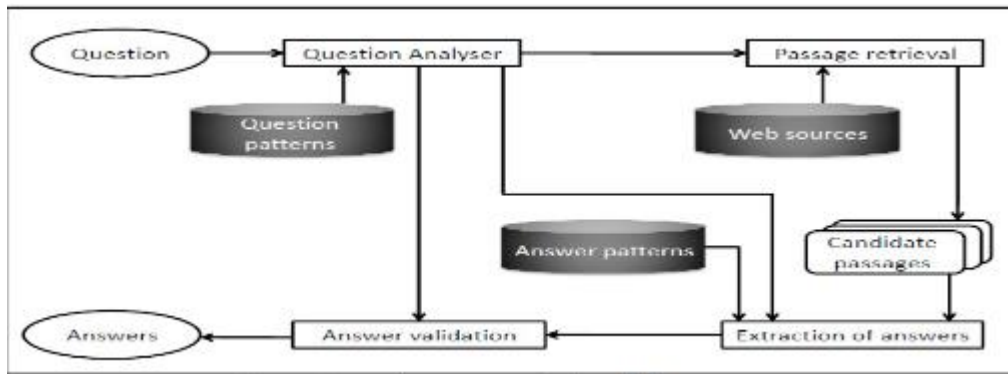
الأجوبة الصحيحة: حيث من أجل كل سؤال من الأسئلة السابقة تم وضع قائمة من الأجوبة الصحيحة المحتملة لهذا السؤال.

باستخدام مجموعة الاختبار السابقة تم الحصول على دقة Precision مساوية لـ 83.3% .

3.1.2.2 نظام QASAL :

لدى نظام QASAL (Brini.W.et al.,2009) بنية متواردة pipeline تتألف من ثلاث مكونات رئيسية وهي: محلل السؤال، المكونة الخاصة باسترجاع الفقرات، مستخرج الجواب.

يتيح هذا النظام طرح أسئلة الحقائق والتعاريف فقط.



الشكل (5) آلية عمل نظام QASAL

تم بناء هذه المكونات باستخدام منصة NOOJ.

منصة NOOJ: هي بيئة لتطوير موارد لغوية (صرف تصريفي، صرف اشتقائي، قواعد محلية وقواعد نحوية ودلالية) وتدعم اللغة العربية، وهي متاحة مجاناً على الموقع.

الإجابة على أسئلة الحقائق:

• تحليل السؤال:

حيث تم وضع بيان (Graph) يشمل القواعد النحوية للأسئلة الممكن طرحها، تم تعريف هذه القواعد في منصة NOOJ، وبالتالي عند ورود سؤال نقوم أولاً وباستخدام NOOJ بتحليل السؤال تحليلاً نحوياً، ثم تمريره على هذه القواعد للقيام بعملية مطابقة أحد هذه القواعد نحصل على بعض

المعلومات عن السؤال وهي نوع الجواب المتوقع، الكلمات الهامة في السؤال question focus وهي أسماء الكيانات الواردة في السؤال، بالإضافة للكلمات المفتاحية.

• المكونة الخاصة باسترجاع الفقرات:

لم ينوه فريق البحث إلى طبيعة نظام استرجاع الفقرات المستخدم في نظامهم، فقط ذكروا أنه مكونة هامة ويعتمد عليها نجاح النظام.

• استخراج الجواب:

يتم أيضاً باستخدام قوالب، حيث باستخدام نظام استرجاع الفقرات، نحصل على كل الفقرات الممكن وجود الجواب فيها، وباستخدام مجموعة من القوالب معرفة مسبقاً في منصة NOOJ ونوع الجواب المتوقع يمكن تحديد الجواب الممكن للسؤال المطروح.

وذلك حيث جرى مسبقاً معالجة الوثائق corpus وذلك بتقطيعها باستخدام المُقطِّع tokenizer الموضح في ورقة العمل (Belguith et al.,2005) ثم ترميزها (tagging) باستخدام محلل صرفي الموصف من قبل (Belguith et al.,2006).

الإجابة على أسئلة التعاريف:

وأخيراً تم تعديل النظام للإجابة على أسئلة التعاريف، أما بالنسبة لأهم التعديلات المدخلة فهي إضافة نماذج أسئلة التعاريف، واستخدام الويب كمصدر للأجوبة وذلك باستخدام محرك بحث Google، واستخدام نماذج خاصة للأجوبة.

الاختبارات والنتائج:

بالنسبة لوثائق الاختبار تم استخدام الويب لأمثلة التعاريف بينما استخدمت مجموعة وثائق corpus مأخوذة من كتب تونسية لأمثلة الحقائق، أما بالنسبة لأسئلة الاختبار فقد تم وضع مجموعة من الأسئلة بشكل يدوي منها 43 سؤال تعاريف حسب ماورد في ورقة البحث.

أما بالنسبة للنتائج فقد ذكرو أنهم حصلوا بالنسبة لأسئلة التعاريف على recall مساوي ل100% و precision مساوي ل94%، بينما لم يتم ذكر أي معلومات عن النتائج التي حصلوا عليها من أجل أسئلة الحقائق.

4.1.2.2 نظام QAS:

اعتمد هذا النظام (kanaa.G et al.,2009) على نفس أفكار نظام QARAB فالاختلافات كانت بسيطة جداً وأهمها تعديل على مستوى نظام استرجاع المعلومات IR حيث جرى تعديل في بنية جداول قاعدة المعطيات الذي سيتم الاعتماد عليه في عملية الفهرسة، بينما بقي النظام يستخدم نفس الخوارزمية وهي نموذج فضاء الأشعة Vector Space Model.

كما تم التغيير أو إعادة بناء بعض الموارد التي كان يستخدمها نظام QARAB في بعض الوظائف مثل عملية التتميط بأنماط الكلام (Part Of Speech) حيث قام نظام QARAB ببناء نظام تتميط خاص به بينما استخدم نظام QAS نظام تتميط جاهز يدعى tagger.

تجزير النظام :

لقد تم بناء هذا النظام باستخدام Visual Basic 6.0 وبيئة تطوير قواعد المعطيات Microsoft Access Database.

الاختبارات والنتائج:

تم مناقشة نتائج نظام استرجاع البيانات فقط في ورقة البحث في حين لم يتم التطرق لنتائج النظام ككل، حيث تم اعتماد الدقة Precision والتغطية Recall.

التغطية:مجموعة الوثائق التي تم استرجاعها وذات العلاقة بالاستعلام / كافة الوثائق.

الدقة: مجموعة الوثائق التي تم استرجاعها وذات العلاقة بالاستعلام / مجموعة الوثائق التي تم استرجاعها.

وتحسب التغطية والدقة من أجل كل استعلام ومن أجل كل مستوى معين للتغطية حيث هناك عدة مستويات للتغطية (11 مستوى) تحسب الدقة باستخدام تابع استيفاء.

وبعد مجموعة الحسابات تم الوصول للقيمة الوسطية لكل من الدقة والتغطية وكانت النتائج كما يلي:

من أجل مستوى تغطية 0،10،20% كانت الدقة مساوية 100% ،أما من عند المستوى 90 و100% كانت الدقة 43%.

بشكل عام لم يضيف هذا النظام أي أفكار أو طرق جديدة في معالجة بعض مشاكل نظم الإجابة على الأسئلة باللغة العربية إنما اكتفى بإعادة إنجاز ماتم التوصل إليه سابقاً في بعض النظم.

2.2.2 نظرة تاريخية إلى التعرف على الكلام AUTOMATIC SPEECH :RECOGNITION

إن الكلام هو الوسيلة الأساسية للتواصل بين الناس. ولعدة أسباب تبدأ من الفضول المعرفي التقني حول تحقيق أنظمة بإمكانيات التحدث البشرية، إلى الحاجة لأتمتة أنظمة وآلات تتطلب تفاعل مباشر وسريع مع الناس، كانت الأبحاث في مجال أتمتة التعرف على الصوت وتحليله (وتركيبه) من قبل الآلة محط أنظار العديد من العلماء عبر العقود الماضية، فالتعرف على الصوت ليس بحثاً جديداً، وإنما حقيقة يملك أسساً تعود لخمسينيات القرن الماضي.

إن تصميم آلة تحاكي في سلوكها السلوك البشري، تحديداً القدرة على التحدث بشكل طبيعي والاستجابة للغة المحكية، شغل العلماء والمهندسين لعقود، منذ خمسينيات القرن الماضي 1950s عندما قام Homer Dudley من مختبرات Bell Laboratories باقتراح نموذج لتحليل الكلام وتركيبه، بدءاً من آلة بسيطة تستجيب لمجموعة صغيرة من الأصوات وصولاً إلى أنظمة معقدة تستجيب للغة الطبيعية المحكية بطلاقة، آخذين بالحسبان التنوع الإحصائي للغة والذي من خلاله يتم توليد الكلام. بالاعتماد على النماذج الإحصائية المتقدمة للكلام في 1980s لاقت أنظمة التعرف الأوتوماتيكي على الكلام انتشاراً واسعاً في مجال التطبيقات التي تتطلب واجهات تخاطبية مباشرة مع الإنسان مثل مراكز الاتصال الأوتوماتيكي على شبكات الاتصال الهاتفية automatic call centers، ونظم المعلومات القائمة على الاستعلام query-based information systems والتي تقوم بخدمات مثل توفير معلومات الرحلات في المطارات والإجابة عن أسعار البورصة و توفير تقارير عن الطقس .. الخ.

نظرة تاريخية لنظم وتطبيقات تعرف على الكلام:

من 1950s وحتى 1960s: يسمى عصر "baby talk"، حيث تم التعرف على الأرقام والأعداد فقط، ففي عام 1952 تم اختراع "Audrey" من قبل Bell Laboratories حيث كان باستطاعته فهم الأرقام فقط، في حين تم اختراع تقنية "shoebox" في عام 1962 وكان قادراً على فهم 16 كلمة باللغة الانكليزية،

ولاحقاً تم تحسين تقنية التعرف على الكلام بحيث أصبحوا قادرين على التعرف على 9 أحرف ساكنة و4 أخرى صوتيات.

في 1970s: ساهمت وزارة الدفاع الأمريكية بشكل كبير في تطوير أنظمة التعرف على الكلام، وفي الفترة ما بين 1971 و 1976 أوجدوا نظام (Speech Understanding Research) "DARPA SUR"، ونتيجة لذلك تم تطوير نظام التعرف "Harpy" من قبل Carnegie Mellon والذي تمتع بقدرته على استيعاب 1011 كلمة. وقد وظف أنظمة فعالة في عمليات البحث المعتمدة على جمل منطقية (and, or ..).

وعلى التوازي مع هذه الأنظمة كان هناك تطوراً في التقنيات المستخدمة في هذا المجال فقد قامت Bell Laboratories بتطوير جهاز قادر على فهم صوت أكثر من شخص معاً.

في 1980s: كان تطوير نموذج hidden Markov model تقدماً مميزاً في هذه المرحلة، حيث مكّنهم من استخدام علم الصوت واللغة وبناء الجمل في نموذج احتمالي موحد.

في 1990s: مع بداية توافر معالجات أسرع أصبحت برمجيات التعرف على الكلام ممكنة أكثر وبجودة أكبر من سابقتها، فقد أصدرت شركة Dragon برنامج "Dragon Dictate" والذي كان أول برنامج في العالم يستهدف المستهلكين العاديين، وفي عام 1997 تم تطويره بحيث يمكنك التحدث بحوالي 100 كلمة في الدقيقة.

وبحلول عام 2001 كانت تقنيات التعرف على الكلام قد دخلت في مرحلة عدم تطور حتى أنشأت شركة Google تطبيق "Google Voice Search" لأجهزة الـ iPhones والذي يستخدم مراكز البيانات data centers لإجراء عمليات هائلة من تحليل البيانات اللازمة لمطابقة استفسارات المستخدمين مع أمثلة فعلية على الكلام. وفي عام 2010 قدمت شركة Google خدمة شخصية للتعرف الصوتي على أجهزة الأندرويد لتخزن بواسطتها استعلامات المستخدمين الصوتية للعمل على تحسين نماذج التعرف لديها والتي أصبحت تشمل عدداً هائلاً من الكلمات باللغة الانكليزية. في حين قدمت شركة Apple تطبيق Siri الذكي الذي يعتمد على الحوسبة السحابية cloud computing أيضاً.

نماذج وطرق وخوارزميات التعرف على الكلام:

○ نماذج ماركوف **HIDDEN MARKOV MODELS**:

تستند أنظمة التعرف على الكلام للأغراض العامة الحديثة general-purpose speech recognition systems على نماذج ماركوف المخفية. وهي نماذج إحصائية تنتج سلسلة من الرموز أو الكميات. وتستخدم HMMs في التعرف على الكلام لأن إشارة الكلام يمكن أن ينظر إليها على أنها إشارة ثابتة تبعاً لمجال معين من القيم (piecewise stationary signal) أو إشارة ثابتة لفترة قصيرة في نطاق زمني قصير (على سبيل المثال، 10 ميلي ثانية).

وتستخدم HMMs في معالجة الكلام لأنه يمكن تدريبها تلقائياً وبسيطة ومعقولة حسابياً ككلفة في الزمن. في التعرف على الصوت يكون خرج نماذج ماركوف المخفية سلسلة ذات n بعد من الأشعة ذات القيم الحقيقية (حيث n عدد صحيح صغير، مثل 10)، هذا الخرج نحصل عليه كل 10 ميلي ثانية. ستتكون أشعة الخرج من معاملات cepstral، والتي يتم الحصول عليها من خلال تحويل فورييه العكسي لنافذة زمنية قصيرة من الكلام واستنتاج الطيف spectrum باستخدام تحويل جيب التمام cosine transform، ثم أخذ المعاملات الأولى (الأكثر أهمية).

كل كلمة (أو فونيم phoneme، في معظم نماذج التعرف على الكلام) سوف تملك توزيعاً غاوسياً مختلفاً، ونموذج HMMs لسلسلة كلمات (جملة) يعبر عنها بتسلسل نماذج HMMs لكل كلمات هذه الجملة، وهذه هي الفكرة الأساسية وراء نماذج HMMs، أما حديثاً فقد كملت نظم التعرف على الكلام عدداً من التقنيات لتحقيق دقة أعلى في التعرف زيادة على ما تم تحقيقه في الطريقة الأساسية لنماذج HMMs، على سبيل المثال، في الأنظمة التي تحتاج للتعرف على عدد كبير من المصطلحات هناك حاجة لفهم الكلمات المعتمدة على السياق context dependency.

○ نماذج التعرف المعتمدة على **DYNAMIC TIME WARPING (DTW)**:

وهي طريقة كانت مستخدمة قديماً للتعرف على الصوت، لكنها استُبدلت بطريقة التعرف المعتمدة على HMMs. حيث DTW هي خوارزمية لقياس التشابه بين سلسلتين تختلفان عن بعضهما بالزمن أو بالسرعة. على سبيل المثال التشابه بين نموذجي مشي في فيديو يمكن أن يُحدّد بالرغم من أن الشخص في أحدهما يسير ببطء وفي الآخر يسير بسرعة. استخدمت DTW في معالجة وتحليل الفيديوهات

والبيانات وكل نمط من الداتا يمكن تمثيله خطياً. والجدير بالذكر أنه تم استخدام DTW في مجال التعرف على الصوت للتعرف على الكلام بسرعات مختلفة.

○ النماذج المعتمدة على الشبكات العصبونية *NEURAL NETWORKS*:

منذ ثمانينيات القرن الماضي استُخدمت الشبكات العصبونية feedforward Neural Networks في العديد من جوانب التعرف على الكلام مثل تصنيف المقاطع الصوتية phoneme classification والتعرف على كلمات منفصلة. لكن غالباً كانت تفشل في التعرف على الكلام المستمر ويعزى ذلك لضعف قدرتها على تتبع السياق. لذلك تم استخدام deep feedforward neural network (DNN) والتي تتكون من عدة طبقات مخفية بين طبقتي الدخل والخرج وتتمتع بقدرتها على نمذجة العلاقات غير الخطية حيث تسمح الطبقات الإضافية بتراكم تركيبات معقدة من المميزات features من الطبقات الدنيا مما يؤدي لإمكانية التعامل مع نماذج معقدة من الكلام خلال عملية التدريب.

أنظمة مشابهة (وكيل تخاطبي لحوارات طلب الطعام):

في نهاية عام 2016 أعلنت شركة القهوة العالمية Starbucks أنها سوف تتضمن لركب الشركات التي تستخدم الذكاء الصناعي وأنظمة الحوار لتلبية أنواع معينة من الخدمات، فقررت أن تقدم تطبيق "My Starbucks Barista" وهو نظام حوار Chabot لطلب القهوة عبر الانترنت. أكدت الشركة أنها قامت بإنشاء متاجر جديدة متخصصة لتخديم الطلبات القادمة من تطبيق الموبايل فقط حيث لا تقوم بالبيع مباشرة للزبائن في هذه المتاجر في إشارة إلى زيادة مبيعاتها.

من التطبيقات المشابهة في هذا المجال Dominos Bot والذي يتيح أيضاً للمستخدمين طلب الطعام عبر أحد منصات الدردشة كالمسنجر.

3.2.2 لمحة عن منهجيات الاختبار المتبعة في نظم الإجابة عن الأسئلة والوكلاء التخابية:

انتشرت في الآونة الأخيرة نظم الإجابة عن الأسئلة بشكل كبير، وتشعبت تقنياتها مما استدعى تنظيم طرق معينة لتقييمها، فكانت بداية ظهور المؤتمرات والمنظمات الخاصة بتقييم نظم الإجابة عن الأسئلة والتي أسهمت بشكل كبير في تنظيم طرق التعامل مع هذا النوع من النظم، وسنستعرض بعض المؤتمرات التي تقوم بعمليات التقييم هذه.

ينعقد كل بضعة سنوات المؤتمر العالمي لاسترجاع النصوص (TREC (Text Retrieval Conference، ومنتدى التقييم عبر اللغات (CLEF (Cross-Language Evaluation Forum واللذان يحتويان على مهام Tasks متعلقة بنظم الإجابة عن الأسئلة وتقييمها، لكنها لا تشمل اللغة العربية، ففي عامي 2001 و 2002 نُظمت ضمن مهام TREC مهمة لاستخراج المعلومات عبر اللغات (CLIR Cross-Language IR) باللغة العربية لكن لم تنظم أية مهمة تخص نظم الإجابة عن الأسئلة بهذه اللغة (QA Question Answering).

أهداف التقييم:

تعتبر شركة TREC من أولى الشركات التي أظهرت اهتماماً كبيراً بنظم استرجاع المعلومات، ونظم الإجابة عن الأسئلة المعتمدة على استرجاع المعلومات، وكانت أهدافها:

- تشجيع الأبحاث في هذه المجالات.
- زيادة التواصل بين الشركات ومخابر الأبحاث.
- تطوير تقنيات نظم استرجاع المعلومات.

تقييم نظم الإجابة عن الأسئلة QA حسب ما ورد في TREC-8 1999:

قدمت شركة TREC من خلال مؤتمرها المنعقد عام 1999 مجموعة من الوثائق (document collection) ومجموعة أسئلة اختبارية (test set of questions) كانت عبارة عن أسئلة حقائق وإجابات قصيرة مثل:

كم عدد السعرات الحرارية في فنجان كبير؟

How many calories are there in a Big Mac?

أين يقع تاج محل؟

Where is the Taj Mahal?

ويضمن وجود وثيقة واحدة على الأقل لكل سؤال ضمن مجموعة الوثائق تحتوي على الإجابة لهذا السؤال، يقوم المشاركون من خلال نظامهم بإيجاد خمسة إجابات مرتبة (ranked) حسب النظام من الأصح إلى الأقل صحة، كل إجابة تسمى response وهي مكونة من ثنائية {document – id}، {answer – string}.

آلية التقييم:

يقوم أعضاء اللجنة بوضع علامة صح أو خطأ لكل إجابة من الإجابات الخمسة ولكل الأمثلة، ثم بوضع نتيجة score لكل سؤال بناءً على ترتيب أول إجابة صحيحة من الإجابات الخمسة التي قام النظام بترتيبها، أي إذا كانت الإجابة الأولى حسب ترتيب النظام صحيحة فيوضع score = 5 لهذا السؤال وإلا ينظر إلى الإجابة الثانية وهكذا حتى الإجابة الخامسة إذا كانت صحيحة فإن score = 1 وإلا يعتبر النظام لم يجب على هذا السؤال ويوضع له score = 0، ونفس العملية تعاد من أجل كل سؤال من الأسئلة الباقية وفي النهاية يحصل المشارك على علامة تقييمه وتعرف بـ score of run وهي عبارة عن متوسط علامات الأسئلة:

$$\text{Score of run} = \frac{\sum_{i=1}^N \text{score of question}(i)}{\text{Number of questions}}$$

هذا فيما يخص نظم الإجابة عن الأسئلة المعتمدة على استرجاع المعلومات من وثائق وقواعد معرفة.

لنرى أهم العوامل التي يُنظر إليها في تقييم نظم الإجابة عن الأسئلة بشكل عام:

- عامل الوقت Timeliness: يجب أن يتم الجواب على السؤال بالزمن الحقيقي مهما كان عدد المستخدمين للنظام.
- الدقة Accuracy: الدقة في الإجابة مهمة لأن الإجابة الخاطئة أسوء من عدم الإجابة عن السؤال.
- صحة الإجابة Completeness: يجب الإجابة الكاملة على السؤال.
- التصرف في حال عدم التمكن من الإجابة.

أما بالنسبة لآليات اختبار الوكلاء التخاطبية فكانت هناك عدة دراسات لتحقيق تقييم هذه الأنظمة أهمها دراسة في عام 2010 قدمت مجموعة من المقاييس التي ينظر إليها عند تقييم نظام حوار، تدرجت هذه

المفاهيم أو المقاييس مع المراحل العامة لمعظم النظم التخاطبية، حيث أسقطت هذه المقاييس على كل من نظام AINI (وهو نظام تخاطبي يعتمد على فهم وتحديد معنى الجملة بغرض الإجابة عنها) ونظام START (وهو أيضا نظام للإجابة عن الأسئلة بالاعتماد على تحليل السؤال وفهمه)، وسوف نتكلم عن هذه المقاييس بالتفصيل لقياسها على نظامنا لتقييم أدائه.

3. الفصل الثالث

التحليل

1.3 المتطلبات الوظيفية:

- القدرة على تحويل الصوت إلى نص مكتوب ومعالجته.
- القدرة على فهم النص (رسالة المستخدم) والرد المناسب عليه.
- القدرة على المحافظة على فهم السياق (والمحصور بمجال طلب الطعام) بشكل تقريبي ضمن المحادثة الواحدة مع الزبون.
- القدرة على تمييز العناصر الرئيسية لإتمام طلبية طعام من مطعم.
- أرشفة الفواتير.
- إمكانية عرض الفاتورة للزبون بعد الانتهاء من الطلب.

2.3 المتطلبات غير الوظيفية:

- زمن الاستجابة Response Time:
في الأنظمة الحوارية يجب أن يكون زمن الاستجابة طبيعي أو قريب من الطبيعي، حتى لا يمل المستخدم من المحادثة بفواصل زمنية عند كل رد يجب أن يرده النظام.
- قابلية إعادة الاستخدام Reusability:
إن أي نظام يمكن أن تعيد استخدامه في مكان آخر بجهد قليل، هو نظام فعال يوفر جزءاً من الكلفة والزمن في حال استخدامه كجزء من نظام آخر، وفي مجال طلب الطعام يجب أن يكون النظام قابلاً لإعادة الاستخدام حتى تتمكن المطاعم على اختلاف مبيعاتها ووجباتها الاستفادة منه.
- قابلية التوسيع Scalability.
- قابلية إضافة مميزات جديدة في المستقبل لتحسين النظام Extensibility.
- الوثوقية Reliability: بحيث يعالج ارتكاب خطأ في فهم الطلبية، عن طريق تأكيد الطلب للمستخدم.
- الأمان والخصوصية Security and Privacy.
- سهولة الاستخدام Ease of Use.
- الجودة Quality.
- التوافقية مع عدة منصات Platform compatibility.

- الإتاحة availability: ترتبط بنظام دوام المطعم المستخدم.

3.3 فئات المستخدمين المستهدفين:

ينقسم مستخدمو نظام طلبات الطعام عبر الانترنت إلى قسمين:

- صاحب المطعم Restaurant Owner: وهو المستخدم الذي سوف يضيف خدمة طلب الطعام الخاص بمطعمه بطريقة حوارية عبر الانترنت.
- طالب الطعام The end user: وهو الشخص الذي سوف يقوم بإجراء محادثة لطلب الطعام من المطعم.

4.3 حالات استخدام النظام:

Use case #1:

Name	Login
Actors	End user
Trigger	
Pre-condition	User should be signed-up.
Basic Flow	<ol style="list-style-type: none"> 1. User opens the Login activity. 2. User enters his email and password. 3. Application examines this information. 4. Application shows the chat activity to the user.
Alternative Flow	<ol style="list-style-type: none"> 4. Application rejects email or password. 5. Application opens the Login activity again.

Use case #2:

Name	Sign-up
Actors	End user
Trigger	
Pre-condition	
Basic Flow	<ol style="list-style-type: none">1. User opens the sign-up activity.2. User enters his email and password and some other information.3. Application examines this information and accept it.4. Application generates user_id to the user and insert it into DB.5. Application shows the chat activity to the user.
Alternative Flow	<ol style="list-style-type: none">4. Application rejects the email because it's already used.5. Application opens the sign-up activity again.

Use case #3:

Name	Ordering & Inquire
Actors	End user, FoodOrdering API
Trigger	
Pre-condition	The user should be Logged-in.
Basic Flow	<ol style="list-style-type: none">1. User sends voice or text message.2. Application sends message with user_id.3. FoodOrdering API gets the user_id and generate an order_id within the first time he/she sends a message in the current session.4. FoodOrdering API sends user_id, order_id and the API result message as a response to the application.5. Application shows the message response to the user.

Use case #4:

Name	Add dish
Actors	Restaurant owner (user)
Trigger	
Pre-condition	Restaurant owner should be signed-up and logged-in.
Basic Flow	<ol style="list-style-type: none">1. User opens the add dish page.2. User Enter the name, size, price and description of the dish.3. Application saves these data into DB.

Use case #5:

Name	Add dish's ingredients
Actors	Restaurant owner (user), FoodOrdering API
Trigger	
Pre-condition	1. Restaurant owner should be logged-in. 2. Restaurant owner should add a dish.
Basic Flow	1. User opens the add dish's ingredients page. 2. User Enter the ingredients of the dish and submits it. 3. Application saves these data into DB and sends it to the FoodOrdering API to use it in extending process of the ontology.

4. الفصل الرابع

التصميم البنوي

1. مقدمة:

في الوقت الذي نتطلع فيه لبناء وكيل تخاطبي قادر على فهم جُمل المستخدم والرد عليها بشكل مناسب لتحقيق هدف محدد هو تسهيل عمليات الطلب "وتحديداً طلب الطعام" عبر الانترنت بدقة جيدة من جهة، وإغناء المحتوى العربي من التطبيقات الداعمة للغة العربية التي تستخدم تقنيات ذكية لتحقيق أهداف كانت في مرحلة ما خيالياً علمياً من جهة أخرى، وزيادة على ذلك تحدي معالجة لغة عربية محكية لا تخضع لقواعد ناظمة كاللغات الرسمية كالعربية الفصحى والانكليزية .. الخ، لم تكن هنالك خيارات واسعة داعمة لنا في دراستنا وتجهيزنا وهكذا نظام سواء من أدوات أو خوارزميات أو طرق مُتَّبَعَة آتت ثمارها وأثبتت جدارتها، مما اضطرنا للخوض في عدة تقنيات وتجربتها عملياً لدراسة جدوى تنفيذها في هذا المجال.

فكانت البداية من استخدام أوتومات كان يُرجى منه أن يكتشف عدة أنماط من الأسئلة في الجمل الواردة لكن عدم خضوع اللغة العامية لقواعد، وتنوع صياغة الجمل تبعاً لكل شخص أو تبعاً لكل تجمع من الأشخاص في أحسن الأحوال، جعل فكرة الأوتومات غير ناجعة.

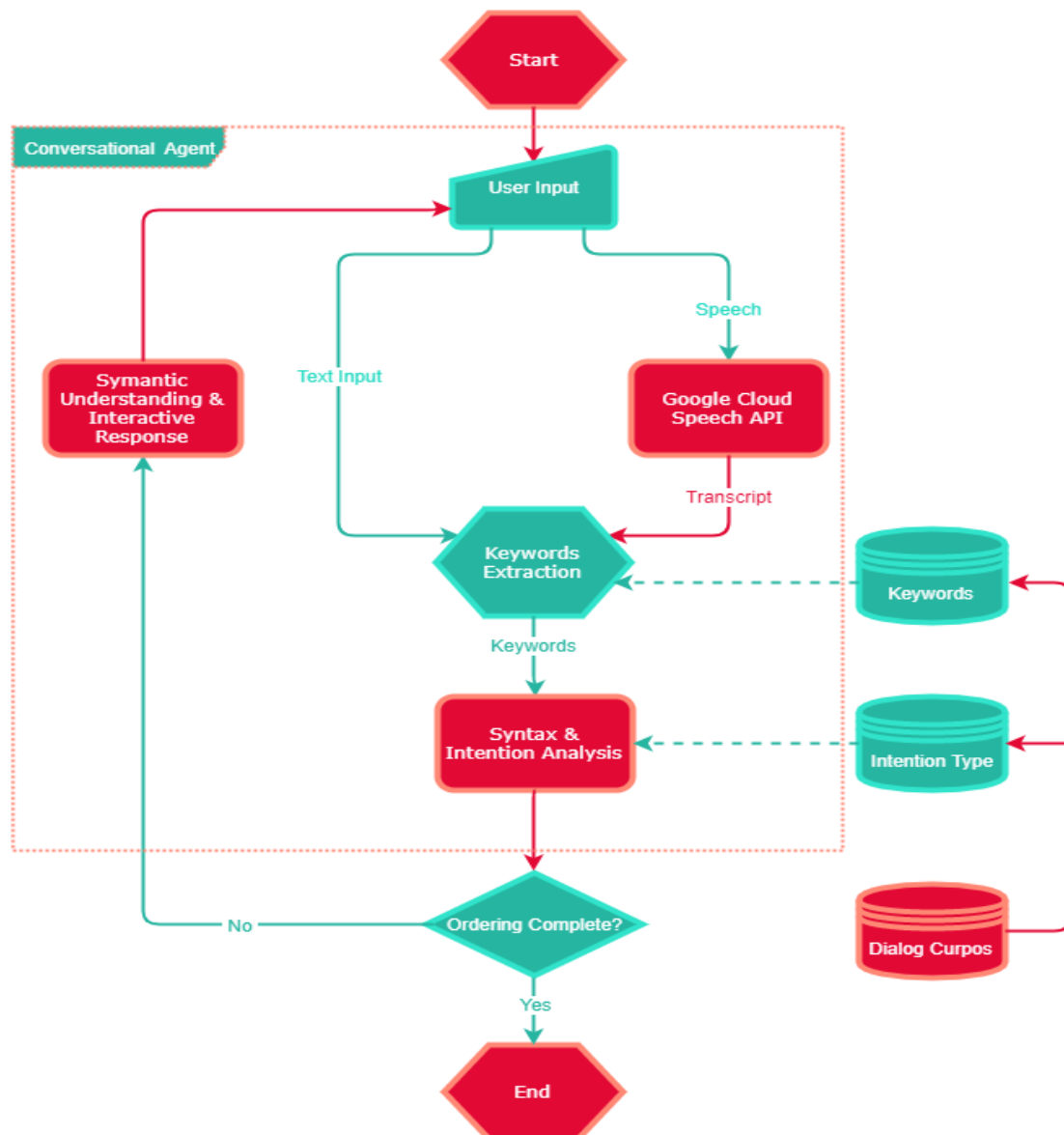
تلتها عدة محاولات في مجال البحث عن كلمات مميزة ضمن الجُمل وربط هذه الكلمات بمعانٍ محددة، وهذا ما اعتمدت عليه Watson chatbot الخدمة التي قدمتها شركة IBM لتدريب محركها الخاص بأنظمة الحوار، حيث قامت بتخزين المحادثات مع المستخدمين وتصنيفها لفهم نية المستخدم من الجملة أو السؤال ومن ثم استخدام هذه البيانات لتدريب chatbot الخاص بها.

وصولاً إلى الاستعانة بأنتولوجيا (أو عدة أنتولوجيا) لتحقيق دقة أعلى في فهم جُمل الدخول والقدرة على التفاعل مع المستخدم ضمن سياق معلوم وهو طلب الطعام.

2. طريقة العمل المقترحة لتحقيق نظام حوار بالغة العربية العامية:

يوضح الشكل (6) بنية النظام المقترحة، ونلاحظ من خلال الشكل أن الوكيل التخاطبي المقترح هو من يلعب الدور الهام في هذه البنية، ولقد قمنا بتحقيق هذا الوكيل باستخدام أربع مهام جزئية: التعرف على الكلام speech recognition، استخراج الكلمات المفتاحية keyword extraction، الحصول على المحتوى الدلالي للكلمات semantic derivation، و الإستجابة التفاعلية interactive response.

المخطط العام لعمل النظام هو كمايلي: يقوم المستخدم بإدخال جملة الطلبية إما صوتياً عبر الميكروفون أو نصياً، ثم يتم التعرف على الكلام باستخدام Google speech API لتحويله لنص، بعد ذلك يتم تمرير النص على وحدة استخراج الكلمات المفتاحية للحصول على الكلمات المفتاحية، هذه الكلمات تستخدم للوصول إلى المعنى الدلالي للدخل وبالتالي فهم وتحديد نمط الجملة، وأخيراً الاستجابة التفاعلية للنظام تقوم بالرد المناسب لإتمام الحوار مع الزبون.



الشكل (6) بنية النظام

3. جمع وتحليل البيانات DIALOG CORPUS:

1.3 جمع البيانات:

بيانات الحوارات التي تم جمعها هي عبارة عن نوعين: حوارات صوتية وحوارات كتابية، وتم استخدامها من أجل عمليات التقييم Evaluation من جهة ومن جهة أخرى من أجل تحليل بنية الجمل ومعرفة الطرق المختلفة التي يتخاطب بها الزبون والمطعم عادةً أثناء طلب الطعام، وتأتي أهمية هذه البيانات كون اللغة المعالجة هي اللغة المحكية التي لا تخضع لقواعد معينة للكلام أو لترتيب معين للكلمات في الجملة.

إن الطريقة المثلى لجمع هذه البيانات هي تسجيل المكالمات الواقعية التي تخص الطلبات في مطعم ما، ولكن لصعوبة تحقيق هذه الطريقة وعدم قدرتنا على إقناع أصحاب مطاعم البيتزا بتسجيل المكالمات، تمت محاكاة المحادثات بين الزبون والمطعم، فطلبنا من عدد من الأشخاص تسجيل مقاطع صوتية تمثل طريقة طلبهم للطعام من مطعم وبعد ذلك تم تحويل المقاطع الصوتية إلى نصوص كتابية والتي بلغ عددها حوالي 100 حوار مختلف، وقد لوحظ أن هذا العدد معقول ليشمل معظم أساليب طلب المستخدم للطعام، وذلك كون النظام متخصص بمجال معين وهو طلب البيتزا وبالتالي الكلمات المستخدمة محدودة نوعاً ما.

2.3 تصنيف الكلمات المفتاحية :

بعد تحليل الحوارات التي تم جمعها، وجدنا بعض أنماط الكلمات التي تلعب دوراً هاماً في طلب الطعام، مثل اسم الطعام ، كمية الطعام(عدد)، حجم الطعام (وسط، كبير، صغير)، وتم تحديد هذه الكلمات وتقسيمها إلى 9 أنماط حسب المعنى كما يلي :

نمط الكلمات المفتاحية	المعنى	مثال
FOOD	أسماء الأطباق	بيتزا الفصول الأربعة
AMOUNT	الكمية المطلوبة من الطعام	واحد، تنتين "اثان"، خمسة، ثلاثة
SIZE	حجم الطبق	عائلي، وسط، كبير
WHAT-PRICE	استفسار عن السعر	شو سعر، بأديش

DESCRIPTION-TERM	استفسار عن توصيف طبق، أو استفسار عن أطباق تملك صفة معينة يحددها المستخدم	شو هي، شو مكونات
WANT	طلب طعام	بدي
INGREDIENT	مكونات طبق	جبنة، لحم
INGREDIENT-PREFERENCE	تفضيلات الطبق وتترافق باسم المكون	بدون ، مع ، بلا
OTHER	الكلمات التي لا تنتمي للأنماط السابقة	زكاتك ، لوسمحت..

الجدول (2) جدول أنماط الكلمات

أما بالنسبة النمط INGREDIENT-PREFERENCE فتم تفصيله وفق الحالات التالية:

- INCREASE-ING: وهذا النمط للكلمات المساعدة التي يستخدمها الزبون للتعبير عن رغبته في زيادة مكون من مكونات الطعام مثال (بدي بيتزا فصول جبنة **زيادة**)
- DECREASE-ING: وهذا النمط للكلمات المساعدة التي يستخدمها الزبون للتعبير عن رغبته في تخفيف مكون من مكونات الطعام مثال (بدي بيتزا فصول و **خفّلي** الجبنة عوشها)
- REMOVE-ING : وهذا النمط للكلمات المساعدة التي يستخدمها الزبون للتعبير عن رغبته في حذف مكون من مكونات الطعام مثال (بدي بيتزا مارغريتا **بدون** درة)

4. نظام التعرف على الصوت:

دور هذا القسم هو تحويل الدخل الصوتي إلى نص كتابي ليتم معالجته لاحقاً، وهو أول معالجة تتم على الدخل، أما في حال لو كان الدخل كتابياً فلا داعٍ لتطبيق هذه المرحلة.

ولتحقيق ذلك قمنا بالاستعانة بخدمة تعرف من غوغل CLOUD SPEECH API التي تمكن مطوري التطبيقات من تحويل الصوت إلى نص مكتوب باستخدام نماذج معقدة من الشبكات العصبونية deep learning neural network، وذلك لعدة أسباب:

○ الحصول على النص المكتوب للكلام transcript في الزمن الحقيقي real time، مع القدرة على الحصول على النص لملف صوتي تم تسجيله مسبقاً.

○ القدرة على التعرف على حوالي 80 لغة محكية (وحتى في بعض الأحيان التمييز بين اللهجات كالعربية الأردنية والعربية الخليجية والعربية اللبنانية .. الخ)، وما تشمل هذه اللهجات من تنوع واسع في المفردات.

○ القدرة على تكييف التعرف على الكلام اعتماداً على مجموعة من الكلمات تناسب محتوى بعينه

Context-Aware Recognition.

○ القدرة على التعرف أثناء الكلام Streaming Recognition.

○ القدرة على معالجة الضجيج الذي يمكن أن يكون مرافقاً للكلام Noise Robustness.

○ القدرة على تحسين دقة التعرف مع مرور الزمن بسبب استخدام تقنيات machine learning.

○ القدرة على تجنب المصطلحات غير المرغوب فيها والتي ربما تؤثر على دقة التعرف

Inappropriate Content Filtering.

○ القدرة على التلاؤم مع أي بيئة وعلى أي جهاز.

ومن أجل الحصول على نتائج جيدة لهذه الخدمة هناك عدة شروط يجب تحقيقها (في حال عدم استيفاء كل الشروط سوق تقل دقة التعرف على الكلام)، من هذه الشروط:

○ معدل الترميز Sampling Rate: أخذ الصوت بمعدل ترميز 16000 Hz، وإلا بتردد المصدر عوضاً عن عمليات resampling لأنها تقلل من الدقة.

○ حجم الإطار Frame size: في حالة التعرف المباشر من المايكروفون Streaming recognition يؤخذ الصوت على شكل إطارات ترسل هذه الإطارات كرسائل متتالية ليتم التعرف عليها وكلما كبر حجم الإطار كان ذلك أفضل كدقة تعرف ولكن ذلك سوف يسبب تأخراً زمنياً، حجم الإطار 100-millisecond وهو مناسب وسطياً للتغلب على مشكلة التأخر الزمني ومشكلة نقص الدقة.

○ يجب أن يكون المايكروفون قريباً بما يكفي وخصوصاً إذا كان هناك ضجيج محيط بالمستخدم.

○ تجنب تشذيب الصوت وتحسينه.

○ كل عمليات إزالة الضجيج يجب إيقافها أثناء التسجيل.

- بالإضافة لنصائح أخرى متعلقة بمستوى الصوت وتردده وصيغة تخزينه، وقد تم تحقيق معظم هذه الشروط باستخدام مكتبة على أجهزة الأندرويد مما أمكن الحصول على دقة معقولة للتعرف. والجدير بالذكر أن نسبة التعرف على الكلام بالنسبة لجمل اللغة العربية كانت جيدة جداً وهو هدفنا في تطبيقنا الموجه لمتحدثي اللغة العربية المحكية .

5. عمليات قبل المعالجة (PREPROCESSING):

دخل هذه المرحلة أحد جمل المستخدم، والتي تكون عبارة عن نصاً كتابياً، وسيتم عليها بعض العمليات:

1.5 التقطيع TOKENIZING:

وهي خطوة هامة في عمليات المعالجة المسبقة preprocessing، وتمثل عملية تقسيم النص إلى سلسلة من الكلمات الفردية أو تسلسل كلمات (n-gram)، واختيار المقطع المناسب يتوقف على اللغة المعالجة، وعلى مسألة البحث، وهنا تم اختيار مقطع بسيط يعتمد على فصل الكلمات عن طريق وجود فراغات بينها فلا يوجد كلمات معقدة أو مؤلفة من أكثر من جزء كما في لغات أخرى.

2.5 إصلاح الدخل REPAIRING:

أثناء تحليل البيانات لوحظ أن الزبائن لا يلتزم بذكر اسم الطبق كاملاً أثناء طلب الطعام، فمثلاً بعض الزبائن يطلب "بدي بيتزا الفصول" ومنهم من يطلب "بدي فصول" وآخرون "بدي فصول أربعة"، والمقصود بها جميعها "بيتزا الفصول الأربعة"، فهدف هذه المرحلة تحويل جميع هذه الأشكال المختلفة لأسماء الأطباق إلى اسم واحد وهو الاسم الأصلي المعروف للطبق.

6. استخراج الكلمات المفتاحية KEYWORDS EXTRACTION :

دخل هذه المرحلة هو جملة الدخل بعد عمليتي Tokenizing و Repairing ، والخرج هو أيضاً جملة ممثلة بالأنماط المقابلة لكل كلمة، مثال لتكن جملة الدخل: "بدي بيتزا الفصول الأربعة عائلي عدد خمسة" يكون الخرج:

بدي	بيتزا الفصول الأربعة	عائلي	عدد خمسة
WANT	FOOD	SIZE	AMOUNT

الهدف من هذه المرحلة استخراج الكلمات المُميزة التي ستساهم في تحديد نمط السؤال، وتم ذلك باستخدام LingPipe API، عن طريق بناء Dictionary للمساعدة في كشف أسماء العلم أو الأنماط المحددة في الجدول (2) سابقاً.

1.6. مقدمة عن LINGPIPE API :

وهي مجموعة أدوات لمعالجة النصوص، وتستخدم للعديد من المهام :

- إيجاد أسماء الأشخاص، الأماكن، والمنظمات (كشف عن أسماء العلم).
- تصنيف نتائج البحث في تويتر تلقائياً.
- اقتراح الهجاء الصحيح للاستعلامات.

وقد صُممت بنية LingPipe لتكون فعّالة، قابلة للتطوير، وقابلة لإعادة الاستخدام، وتحقق الخدمات التالية:

- Java API مع الكود المصدري source code، بالإضافة إلى وحدات لإجراء عمليات الاختبار الموافقة للكود البرمجي.
- نماذج متعددة اللغات والمجالات Multi-Lingual، Multi-domain .
- التدريب وفق بيانات جديدة ومن أجل مهام جديدة.
- اختيار أفضل النتائج بطريقة إحصائية.

2.6 كشف أسماء الكيانات (NER) NAMED ENTITY RECOGNITION:

وهو عملية العثور على دلالات لأشياء محددة في النص المعالج، مثال: كيانات الأخبار

News Entities: الناس People، المنظمات Organizations، المواقع Locations.

فيمكن مثلاً لمتعرف بسيط على أسماء الكيانات NER أن يكشف أن John جون: person والولايات المتحدة: Location في جملة "يعيش جون في الولايات المتحدة".

1.2.6 كشف أسماء الكيانات في LINGPIPE:

نموذج كشف أسماء البيانات في LingPipe ينطوي على التدريب من نوع Supervised لنماذج إحصائية أو أساليب أكثر مباشرة مثل مطابقة القاموس Dictionary أو مطابقة regular expression وقد تم تصميم كل هذه الأساليب للعمل معاً بسلاسة.

يذكر من أساليب LingPipe للكشف عن أسماء الكيانات:

- Rule-Based Named Entity Detection:

هذا الأسلوب يسهل من عملية كشف أسماء الكيانات، وذلك إذا أمكن كتابة تعبير منتظم Regular Expression يقود للكشف عن الكيان في الجملة، أشهر مثال هو كشف "عنوان الإيميل" والتي يشترط فيها أن يكتب بطريقة معينة وفق قيود لذلك يمكن تمثيله ب RE ، أما كلغة محكية فلا يمكن أن تخضع لقواعد محددة.

- Exact Dictionary-Based Chunking:

هذا الأسلوب الذي يتم استخدامه في التطبيقات التي يكون فيها من السهل تجميع قائمة من الأسماء وأنماطها، فمثلاً "50 سينت" أو اسم منتج مثل "xyx120 dvd player" يمكن أن يكون من الصعب جداً العثور عليه في النص باستخدام NER القائمة على القواعد العامة Rule-Based أو الإحصائية، ويعتمد Exact Dictionary-Based Chunking في عمله على وجود قاموس مساعد، و إجراء عملية Chunking حيث يقوم الـ Chunker باستخراج المقاطع Chunks اعتماداً على المطابقة التامة مع محتويات القاموس.

- Approximate Dictionary-Based Chunking:

يشبه الأسلوب السابق مع اختلاف أن الأول يختار التطابقات التامة بين مقاطع الجملة ومحتوى القاموس أما في هذا الأسلوب فيتم اختيار التطابقات التقريبية ليقبل التطابقات بعتبة مسافة محددة مسبقاً.

- Running a Statistical Named Entity Recognizer :

يستخدم في حال وجود نموذج مدرب.

و تبين أن الطريقة الملائمة لكشف أسماء الكيانات في مثل هذه الأنظمة هي

Approximate Dictionary-Based Chunking وتتم كما يلي:

• بناء القاموس DICTIONARY:

ويتم ذلك عن طريق إدخال بعض المعلومات، مثال:

```
MapDictionary<String> dictionary = new MapDictionary<String>();  
dictionary.addEntry(new DictionaryEntry<String>("50 Cent", "PERSON"));
```

نلاحظ أن المعامل الأول هو الكيان، والمعامل الثاني النمط الموافق، وبناء هذا القاموس معتمد بشكل مباشر على الجدول (2) جدول أنماط الكلمات.

بعض الأمثلة التي تم اعتمادها في النظام لبناء ال Dictionary:

<مارغريتا>	<FOOD>
<مارغريتا مع ذرة>	<FOOD>
<مرتديلا وقطر>	<FOOD>
<ببروني مع الخضار>	<FOOD>
<هوت دوغ مع القطر>	<FOOD>
<قطر مع الدجاج>	<FOOD>
<محشية الأطراف>	<FOOD>
<سلامي مع الخضار>	<FOOD>
<صغير>	<SIZE>
<كبير>	<SIZE>
<وسط>	<SIZE>
<عائلي>	<SIZE>
<حق>	<PRICE>
<بأدين>	<WHAT_PRICE>
<شو حق>	<WHAT_PRICE>

• بناء DICTIONARY-BASED CHUNKER:

إن مهمة هذا النوع القائم على القاموس هي أن يقوم بالمطابقة التقريبية فلا يختار فقط التطابقات التامة، وإنما جميع التطابقات ضمن عتبة مسافة محددة، يتم بناؤه من ال dictionary الذي تم بناؤه في المرحلة السابقة، ومقطّع Tokenizer مع تحديد مسافة تعبر عن العدد الأقصى المسموح به لعمليات الحذف أو الإضافة أو التبديل في أحرف المقطع للحصول على التطابق بينه و بين أحد محتويات القاموس.

3.6 التجذيع STEMMING:

وهو إزالة جزء من السوابق واللواحق وبعض الحروف غير الأصلية حتى يتم إيجاد الكلمات التي لها نفس الجذع عند البحث عن أي منها، وهو غير التجذير Lemmatization حيث أن إعادة الكلمة لجذرها يضيع جزء كبير من المعنى مثلاً "النفسية" و"المنافسة" لهما نفس الجذر.

تجذيع الكلمات غير المفتاحية:

بعد انتهاء عملية الكشف عن أسماء الكيانات اعتماداً على **dictionary-based chunker**، وحسب الجدول (2)، تبين أن بعض الكلمات من النمط OTHER لم يقوم ال Chunker بمطابقتها مع أي من محتوى القاموس، رغم أهمية وجودها في السؤال مثل "أنواع" و"أحجام"، والتي تأتي ضمن السياق بأشكال مختلفة، لذلك لم يكن من المفيد وضع جميع أشكال ورودها ضمن القاموس، وبالتالي تأتي أهمية التجذيع في توحيد تلك الأشكال المختلفة في لفظ واحد، ليتم إعادة عملية المطابقة بالبحث مجدداً عن أنماط الكلمات الضرورية في تحديد آلية الإجابة.

لنفترض المثال التالي "شو أحجام بيتزا الفصول الأربعة؟" بعد تطبيق مرحلة استخراج الكلمات المفتاحية ينتج أن الجملة تحوي (FOOD, DESCRIPTION-TERM) دون الأخذ بعين الاعتبار وجود كلمة "أحجام"، وبتطبيق عملية التجذيع على الكلمات من نمط OTHER أي كلمة "أحجام" تصبح "حجم"، ثم تعاد عملية استخراج الكلمات المفتاحية مرة أخرى لينتج النمط SIZE بالإضافة للأنماط السابقة وبالتالي فهم السؤال على أنه استفسار حجم عوضاً عن فهمه على أنه استفسار عن توصيف بيتزا الفصول الأربعة نظراً لتجاهل "أحجام" في حال عدم تطبيق التجذيع.

7. تحليل نمط الدخول INTENTION ANALYSIS:

إن معرفة نية الزبون وما يريد شيء مهم جداً في أنظمة الحوار، أيضاً التحليل اللغوي يقدم معلومات مهمة عن الجملة ليقرر فيما إذا كانت جملة مقبولة أم لا، وهذه المرحلة تتم عادة بعد تحديد نية الزبون Intention، والمقصود بالنية معرفة هدف المستخدم من الجملة التي يذكرها في الحوار، هل يستفسر عن شيء ما، أم يطلب أم يرحب فقط.

1.7 أنماط جمل الزبائن :INTENTION TYPES

بعد تحليل البيانات وجد أنه يمكن تقسيم جمل الحوار إلى عدة أنماط حسب المعنى والهدف المراد من الجملة، كما يلي:

- **S-WHAT**: الجملة التي تحوي كلمة من نمط WHAT من الأنماط المحددة في الجدول (2) المذكور سابقاً ، تكون تابعة لهذا النوع وتحمل هذه النية، وتدل على أن الزبون يريد أن يسأل عن توصيف شيء ما.

مثال : (شو مكونات بيتزا الفصول الأربعة؟)

- **S-PRICE**: الجملة التي تحوي كلمة من نمط PRICE ، تكون تابعة لهذا النوع و تدل على أن الزبون يستفسر عن سعر طبق ما، مثال (شو سعر بيتزا الفصول الأربعة؟)

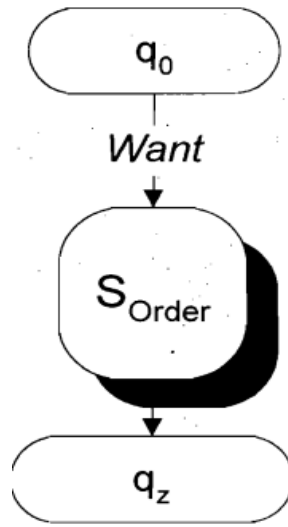
- **S-WANT**: الجملة التي تحوي كلمة من نمط WANT ، تكون تابعة لهذا النوع وتدل أن الزبون يطلب طبق ما، مثال (بدي أربعة بيتزا مارغريتا وسط).

- **S-HAVE**: الجملة التي تحوي كلمة من نمط HAVE، تكون تابعة لهذا النوع وتدل أن الزبون يسأل عن وجود طبق ما، مثال (عنكن بيتزا فصول أربعة؟).

2.7 التحليل اللغوي : SYNTAX ANALYSIS

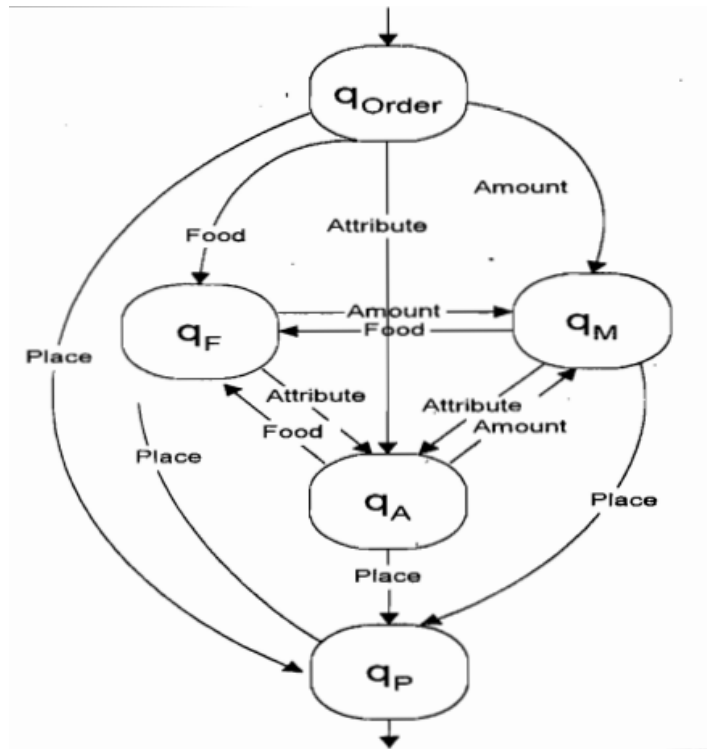
يعتبر التحليل اللغوي مرحلة ضرورية في أنظمة الحوار، فبعد تحديد نمط الجملة، ومعرفة ماذا يريد المستخدم أيريد أن يطلب أم يستفسر، تأتي هذه المرحلة بهدف الكشف عن الجمل الغير مقبولة في نظام الحوار.

ومن أحد الطرق المستخدمة لهدف التحليل اللغوي، وصف جمل الطلب وفق مخطط حالات الانتقال State transition، وفيه تكون الروابط بين الحالات هي عبارة عن أنماط الكلمات المحددة في جدول الأنماط ويتم الانتقال بين الحالات على أساس وجود أنماط الكلمات في الجملة، ولكل نمط من الجمل مخطط خاص به، الشكل التالي سنعرض مثلاً بنية مخطط الحالة لجمل الطلب،



الشكل (7) مخطط الحالة للطلب

الذي يتوضح بوجود كلمة "بدي" "Want" في المخطط، ثم وجود العنصر Sorder والذي يتم فيه تفصيل حالات ورود أنماط الكلمات ضمن جمل الطلب، فمثلاً يكون :



الشكل (8) تفصيل مخطط الحالة للطلب

يعالج هذا المخطط الأشكال المختلفة لورود أنماط الكلمات في جملة، وفي حال أحد الجمل لم تحقق المسار ولم تصل لحالة نهائية ، تعتبر الجملة غير صحيحة لغوياً ويطلب من المستخدم إعادة الإدخال.

وعلى غرار ذلك، قمنا بإحصاء الأشكال المختلفة للجمل العامية وقمنا برسم مخططات الحالة التابعة لكل نمط، ونظراً لطبيعة اللهجة العامية وعدم وجود قواعد تحكم ترتيب الكلمات، توصلنا أن النظام لا يستدعي وجود تلك البنى المقترحة، التي لاتضيف شيئاً على تحليل النصوص ، والتي لم تحسن من طريقة التحكم بالجمل المعالجة.

8. الاستجابة التفاعلية INTERACTIVE RESPONSE:

إن دخل هذه المرحلة يتمثل بنمط السؤال وبعض أنماط الكلمات المستخرجة، مثال "بدي أربعة بيتزا مارغريتا وسط"

تحتوي الدلالات التالية: "S-WANT" + "AMOUNT" + "FOOD" + "SIZE" وهو خرج المرحلة السابقة، والآن يمكن تحديد الإجابة المحتملة حسب نوع السؤال، وسنحدد بعض المصطلحات تستخدم في جمل الإجابة:

- F/D ? : اسم الطعام أو الشراب المذكور في جملة الطلب.
- Description (F/D): توصيف الطعام أو الشراب المذكور.
- Price (F/D): سعر الطعام أو الشراب المذكور.

ومن الواضح أن كل من Description (F/D) و Price (F/D) عبارة عن معلومات متعلقة بالطبق يمكن أخذها من قواعد البيانات.

1.8 إجابة من نمط S-WHAT:

إذا كانت دلالة جملة الدخل من النمط S-WHAT فإن الإجابة تتم وفق الشكل :

Description (F/D) هي F/D ?

فمثلاً لو سأل الزبون (شوهي بيتزا الفصول الأربعة؟) فسيكون الجواب (بيتزا الفصول الأربعة هي "توصيف البيتزا") مع العلم أن توصيف بيتزا الفصل الأربعة يمكن الحصول عليه بسهولة ،موجود في قاعدة المعطيات التي قمنا ببنائها وحددنا فيها لكل طبق سعره وتوصيفه.

2.8 إجابة من نمط S-HAVE:

إذا كانت دلالة جملة الدخل من النمط S-HAVE فإن الإجابة تتم وفق الشكل :

(نعم/لا) يوجد F/D ?

مثلاً لو سأل الزبون (في عنكن بيتزا الفصول الأربعة؟) فسيكون الجواب (نعم ، يوجد بيتزا الفصول الأربعة) .

3.8 إجابة من نمط S-PRICE:

إذا كانت دلالة جملة الدخل من النمط S-HAVE فإن الإجابة تتم وفق الشكل :

سعر F/D ? هو Price (F/D)

مثلاً لو سأل الزبون (شو سعر بيتزا الفصول الأربعة؟) فسيكون الجواب (سعر بيتزا الفصول الأربعة هو "السعر") .

4.8 إجابة من نمط S-WANT:

وهي النوع الأعقد من أنماط الجمل، ولمعالجة هذا النمط تم اقتراح أن الحوار ينتهي بين الزبون "الذي يقوم بالطلب" وبين المطعم عندما يتم الحصول على 3 أنماط للكلمات من طلبه، هي FOOD, SIZE, AMOUNT ، وبالتالي إذا لم توجد هذه الأنماط جميعها في جملة الزبون، يسأل النظام الزبون عن العناصر الموافقة لهذه الأنماط، واحداً تلو الآخر ، لنفترض كمثال الحوار التالي :

الزبون: مرحبا، بدي اطلب بيتزا
النظام: مانوع البيتزا؟
الزبون: بيتزا الفصول الأربعة
النظام: ماالحجم المطلوب؟
الزبون: عائلي
النظام: مالكمية المطلوبة؟
الزبون: عدد خمسة

معالجة هذا الحوار التفاعلي تعتمد على أن النظام يكتشف ماهي الأنماط الغير موجودة في جملة الدخل، ويقوم بتوليد استفسار يطلب من خلاله من الزبون إدخال النمط المطلوب.

ولاينتهي الحوار حتى يتم اكتمال وجود الأنماط الثلاثة المطلوبة في جملة الطلب والتي تمثل خصائص متعلقة بالطعام ،الجدول (3) يوضح ذلك.

AMOUNT	SIZE	FOOD	
			الزبون
		يسأل عن نوع الطعام؟	النظام
		بيتزا الفصول الأربعة	الزبون
	يسأل عن الحجم		النظام
	عائلي		الزبون
يسأل عن الكمية			النظام
عدد خمسة			الزبون

جدول (3) توضيح كيفية عمل نظام الإجابة التفاعلي

عندما يكتمل الحوار، يكرر النظام الطلب للزبون متضمناً اسم الطلب وخصائصه بالإضافة للسعر الموافق، ليتأكد أن طلب الزبون تم فهمه كاملاً، مع إتاحة تعديل الطلب في حال لم يوافق الزبون على الطلب أو أراد تعديل طلبه، مما يحسن من دقة النظام في فهم الطلبات.

الانطولوجيا (ONTOLOGY):

1. تعريف الأنطولوجيا (ONTOLOGY):

مصطلح انطولوجيا له تاريخ طويل في مجال الفلسفة، ويتضمن دراسة الموجودات أو ما نفترض أنه موجود من أجل الوصول المقنع أو القاطع إلى الحقيقة، ويتمثل الاستخدام الأوسع للمصطلح في التمثيل أو التوصيف الرسمي لما هو حقيقي وواقعي بالنسبة للإنسان، كما أن الانطولوجيا يمكن أن تكون ببساطة أشبه بمكنز مع تعريفات لمصطلحاته أو أنها بمثابة تصنيف هرمي معقد جداً للمفاهيم والفئات أو بمثابة حل تقني للمشكلات ذات الطابع الدلالي Semantic-based problems لتقاسم المعلومات كما أن الانطولوجيا تعمل على تعريف طبيعة الحقيقة من خلال تعريف أو تحديد المفاهيم، والكيانات، والمصطلحات والفئات في مجال معين من أجل نمذجة العلاقات بينها، وقد أنشئت الانطولوجيا بهدف الحد من الغموض المفاهيمي والدلالي وجعله في أدنى حدوده في سياق بيئة المعلومات والبيئة التقنية.

وفي مجتمع علم المعلومات استخدمت الانطولوجيا للدلالة على كتل البناء blocks building التي تستخدم لمساعدة الحاسبات والبشر في المشاركة في المعرفة.

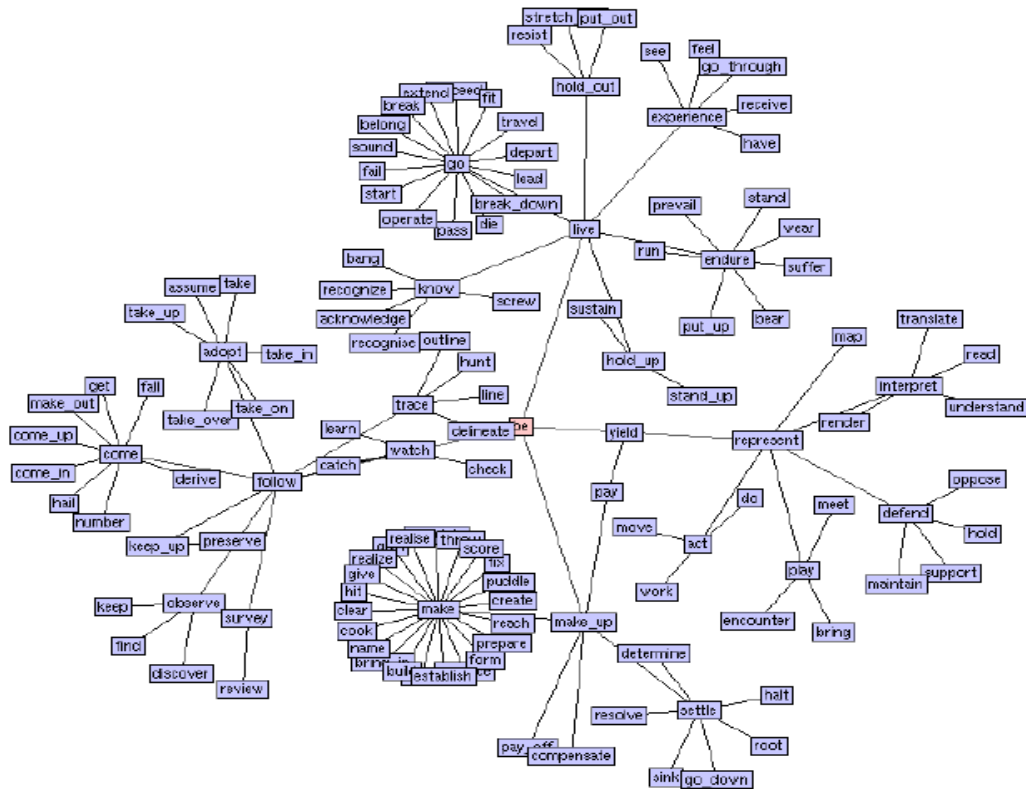
تعد دراسة Gruber من أولى الدراسات التي أرست هذا المصطلح والتي نشرت عام 1992م وقد عرفه على أنه نمذجة تسعى إلى التمثيل المفاهيمي للمجالات والقطاعات المعرفية ، وما تمثله من معرفة فرعية وما بداخلها من كيانات وعلاقات تربط بن هذه الكيانات .

كما عرفها قاموس Harrods للمكتبات على أنها "المجال الذي يعمل توفير مجموعة مشتركة من المصطلحات والمفاهيم وتحديد العلاقات بينهما بدقة شديدة من أجل توصيف مجال موضوعي أو معرفي محدد بطريقة تمكن نظم الحاسبات من الاتصال والتواصل مع بعضها البعض بحيث تكون بمنأى عن نظم التشغيل الفردية ومعماريات المعلومات ونطاقات البرامج.

وتعرف أيضاً على أنها طريقة لتمثيل المفاهيم من حولنا وذلك عن طريق لربط بينها بعلاقات ذات معنى، مما يساعد على فهم أوسع للمفاهيم المختلفة، بهذه الطريقة نستطيع جعل جهاز الحاسب الآلي يصل لمستوى فهم وإدراك للمعاني قريب من فهم وإدراك الإنسان. فتعد الأنطولوجيا بمثابة حلقة الوصل بين فهم الآليات وفهم البشر حيث تستطيع الآلة أن تفسر وتعالج البيانات في إطار من المنطقية والاستدلال من خلال ما ينتج عن توصيف العلاقات والمفاهيم والمصطلحات.

وأشهر مثال على الأنطولوجيا هي ووردنت WordNet وهي قاعدة بيانات معجمية للغة الانكليزية، تجمع الكلمات الانكليزية ذات المعنى الواحد إلى مجموعات من المرادفات وتوضح العلاقة الدلالية بين تلك المجموعات.

فيما يلي شكل توضيحي عن WordNet ontology



الشكل (9) WordNet Ontology

2. مكونات الأنطولوجيا (ONTOLOGY):

1. الكيانات **Entities**: والتي تعرف أيضاً بالمفردات Individuals وتعد المكون الاساسي في بيئة الانطولوجيا وتمثل المستوى الأول داخل الانطولوجيا وتشير كلمة المفردات إلى مختلف الكيانات المادية والمجردة.
2. المفاهيم **Concepts**: أو ما يسمى الفئات Classes وتعد المكون الثاني في بيئة الانطولوجيا وتشير إلى التصنيف الاساسي في مجال ما حيث تشمل على مجموعة من المفردات Individuals أو الكيانات Object والتي تجمع صفات مشتركة جعلتهم ينتمون إلى هذه الفئة التي تم تسكينهم فيها، ولا يقتصر اشتمال الفئات على المفردات فحسب بل تمتد لتشمل فئات فرعية Subclasses لتكون بذلك فئة رئيسية تشمل على فئات فرعية في تسلسل هرمي وبالتالي فإن أعضاء الفئة الفرعية ينتمون بحكم المنطق إلى الفئة الرئيسية.

3. **الخصائص Properties:** والتي تعرف أيضاً بالسمات Attribute توصف كل من الفئات Classes والمفردات Individuals في الانطولوجيا وفق الخصائص المميزة لها عن غيرها والمحددة لذاتها ، ولا يقتصر أمر السمات على توصيف المفردات والفئات بل يمتد ليقوم بتوصيف العلاقات التي تربط هذه الكيانات كلها مع بعضها البعض ، حيث يقوم بتحديد طبيعة ونوع العلاقة التي تربط مفردة بمفردة وفئة بفئة ومفردة بفئة .

4. **العلاقات Relationship :** تعد العلاقات أحد أهم السمات التي تميز الانطولوجيا حيث تتسم العلاقات في بيئة الانطولوجيا بأنها تتمتع بالتوصيف والمسميات والدلالات الأمر الذي يكفل لأنظمة الحاسب الآلي تحقيق التكامل المعرفي بين الكيانات المختلفة.

3. أهداف الأنطولوجيا (ONTOLOGY):

إن أهم هدف للأنطولوجيا هو تنظيم وتصنيف المعرفة لكي تتكامل مع بقية التطبيقات وقواعد البيانات والبرمجيات الأخرى ، وبالتالي تتم عملية الاسترجاع بشكل دقيق وسريع ودقيق.

4. أدوات بناء الأنطولوجيا (ONTOLOGY) :

هناك مجموعة من الأدوات التي يتم الاعتماد عليها في بناء الأنطولوجيات في البيئة الرقمية ، والتي تسمح بإنشاء وبقاء وتصدير الأنطولوجيا لمجال معين ، وتختار منها ما يأتي:

• أداة بروتجيه Protégé:

وهي أداة مفتوحة المصدر ومجانية تعمل على بناء الأنطولوجيا والمعرفة المبنية على إطار العمل، وتعمل هذه الأداة بعد تنصيبها على جهاز الحاسب الشخصي بلغة البرمجة جافا وتتيح للجميع التواصل كمستخدمين للأنطولوجيا ويقومون بتطوير تطبيقاتها وتبادل الخبرات من خلالها ، كما أنها بيئة عمل واسعة ومرنة قابلة للتطوير بسهولة ، ويدعم هذه الأداة مجموعة من الخبراء "مطورين وأكاديميين وغيرهم" ممن يستخدمونها كحلول معرفية في مختلف المجالات كالطب والأحياء وغيرها.

• أداة هوزو Hozo:

تم تطوير بنية أداة هوزو Hozo لبناء محركات بحث دلالية باستخدام الانطولوجيا التي تقوم ببنائها، وتم تطوير هذه الأداة في اليابان بواسطة شراكة بين قسم إدارة نظم المعلومات في جامعة أوساكا وشركة Enegate Co Ltd ، وتعمل هذه الأداة بعد تنصيبها على الجهاز الشخصي ،

ومن وظائفها أنها تقدم منهجية عمل الأنطولوجيا بطريقة منظمة وواضحة بهدف تفعيل دور الأنطولوجيا كواحدة من تقنيات الويب الدلالي وتدعم هذه الأداة نظام الحاسب الآلي لتصميم الأنطولوجيا ، وتطوير بيئة العمل للمبرمجين الذين يعملون على تطوير الأنطولوجيا باستخدام .AP-plications

- أداة بناء الأنطولوجيا (SWOOP):

وهي عبارة عن أداة لإنشاء وبناء الأنطولوجيا ، أنتجتها جامعة University of Maryland, College Park وبعد ذلك تم إتاحتها مجاناً كأداة مفتوحة المصدر على الانترنت للجميع .

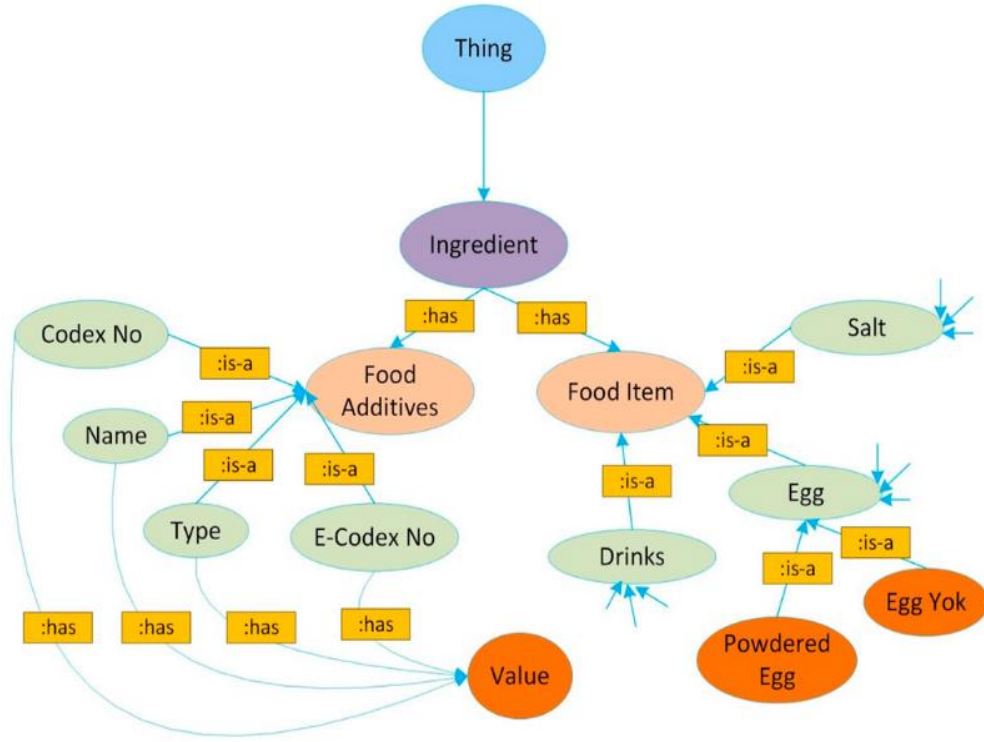
- أداة بناء الأنطولوجيا (OBO):

صممت هذه الأداة في معهد بيركلي للمشاريع مفتوحة المصدر في الولايات المتحدة الأمريكية عام 2007م وتتميز بأنها يتوفر واجهة تعامل سهلة لعرض وبناء الأنطولوجيا ، وتوفر واجهة التعامل درجة مهنية عالية للمهتمين بعلم الأحياء ، حيث تعمل على تحرير وإنشاء الأنطولوجيا بشكل سريع ودقيق بالتركيز على الصفوف والسمات والصفات التي تربط بينها .

5. الأنطولوجيا في أنظمة الحوار:

إن عملية بناء نظام حوار إما أن تتخصص في مجال محدد أو أن تكون عامة، و يجدر الذكر أن بناء مثل هذه الأنظمة يكون أكثر تعقيداً كلما أصبحت عامة أكثر، ونظراً لكون أنظمة الحوار تسعى لأن تكون عامة حتى في المجال الواحد (كنظام الحوار المتخصص في طلب البييتزا يسعى لأن يصبح عاماً من أجل مختلف أنواع الأطعمة ليكون قابلاً للتطبيق ضمن شريحة أوسع من المطاعم)، ولذلك تكون الأنطولوجيا حلاً مساعداً يقدم المرونة للنظام ويخفف من تعقيده ويساعد على تطويره وجعله معممأ أكثر.

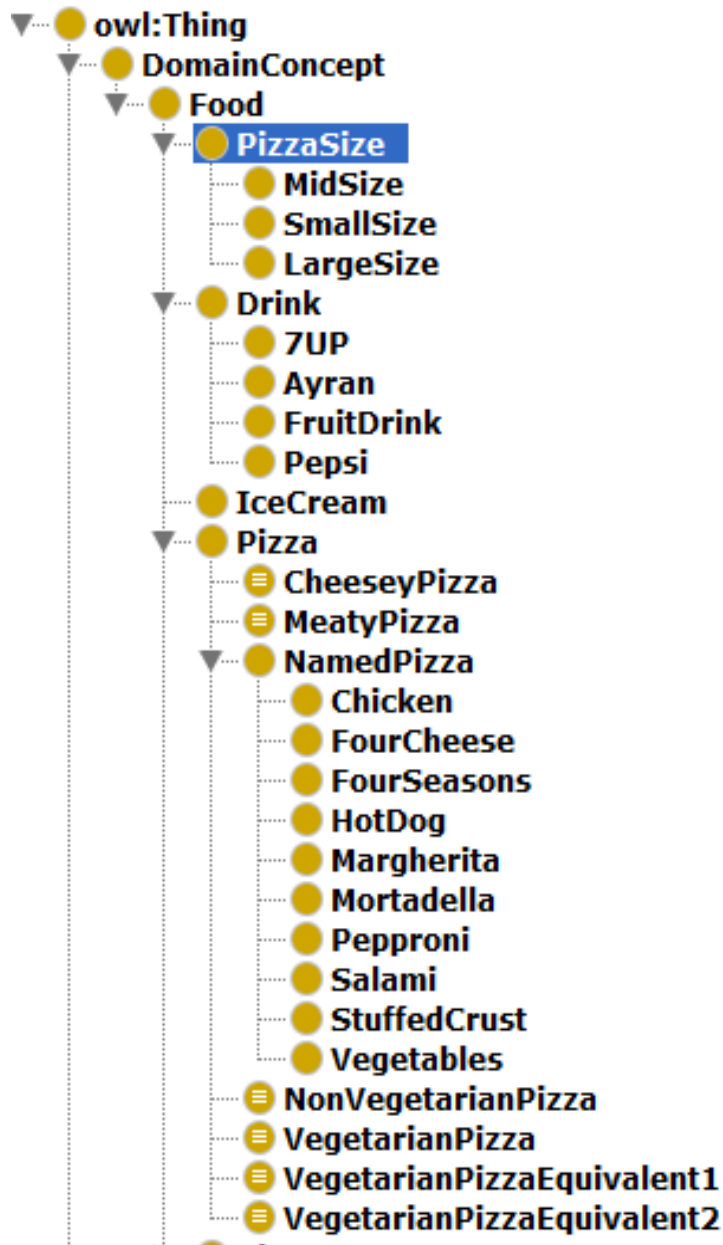
و فيما يلي شكل يوضح أنطولوجيا طعام بسيطة، وبعض العلاقات بين المكونات المختلفة فيها.



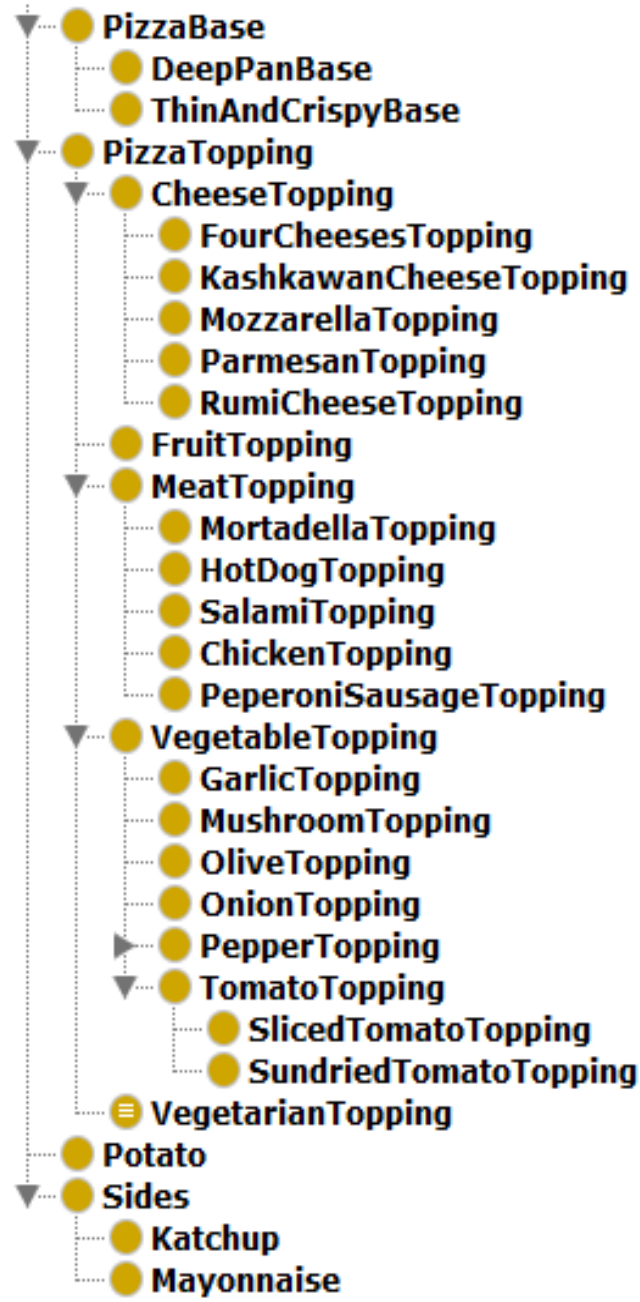
الشكل (10) مثال عن أنطولوجيا طعام

1.5 بناء الأنطولوجيا:

تم بناء الأنطولوجيا من خلال برنامج Protégé المذكور سابقاً، وذلك بإدخال معظم أسماء أطباق البيتزا، ومكوناتها وخصائصها، ويوضح الشكلان (11)، (12) الأنطولوجيا المبنية.

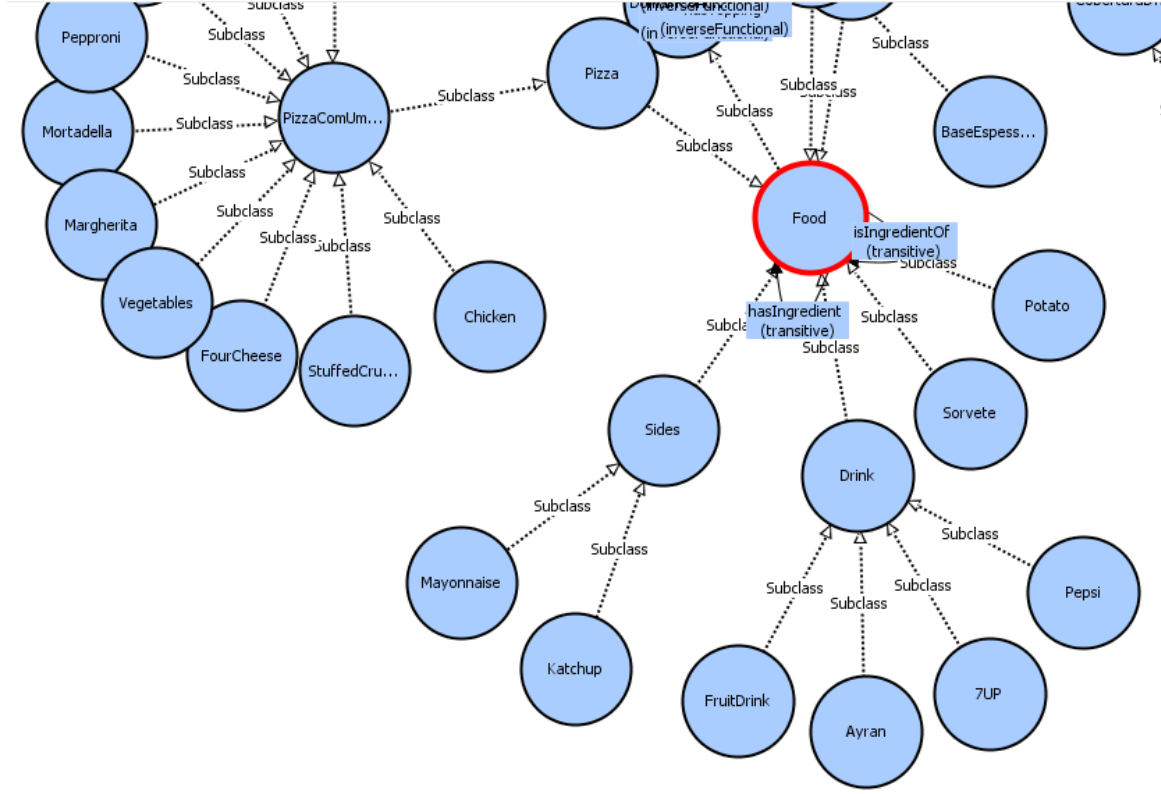


الشكل (11) الأنطولوجيا المساعدة في نظام الحوار ج 1



الشكل (12) الأنطولوجيا المساعدة في نظام الحوار ج2

والشكل التالي يوضح جزء من نموذج الأنطولوجيا المبني وفق مكتبة VOWL وهي مكتبة توفر الرسومات البيانية لعناصر الأنطولوجيا:



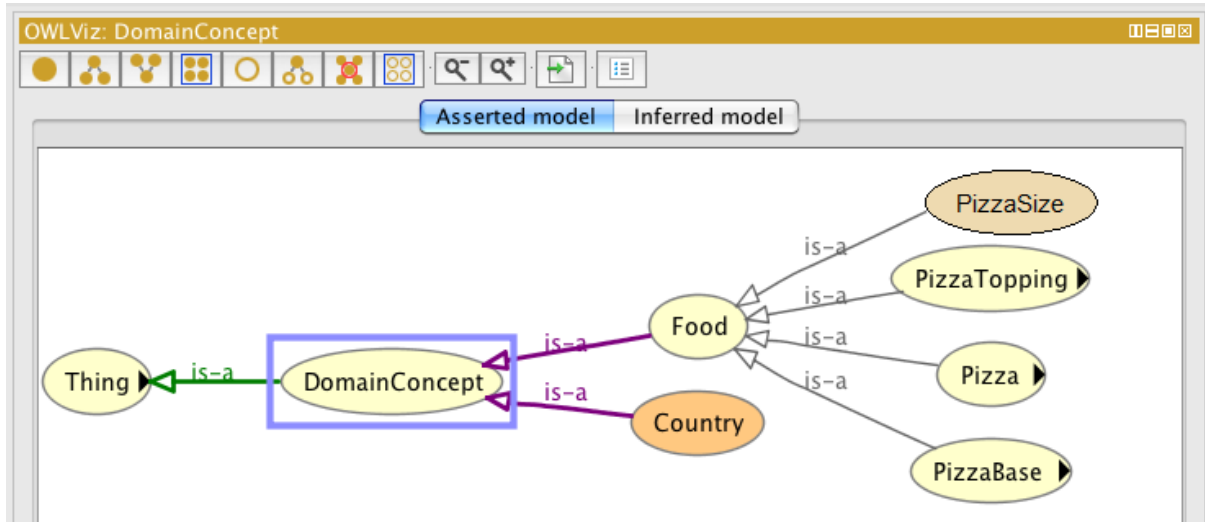
الشكل (13) جزء من نموذج الأنطولوجيا مرسوم وفق مكتبة VOWL

وتحتوي الأنطولوجيا المبنية بشكل عام المكونات التالية:

1. **المفاهيم Concepts:** وهي بمعنى آخر الصفوف Classes وما تشمله بداخلها من صفوف

فرعية Subclasses. وقد تم تعريف المفاهيم الأساسية التالية:

- Pizza ○
- PizzaBase ○
- PizzaTopping ○
- PizzaSize ○



الشكل (14) جزء من المفاهيم الأساسية في أنطولوجيا النظام

2. الأفراد **Individuals**: والتي تمثل نسخ من المفاهيم (Instances).

3. الخصائص **Properties**:

وهي التي تربط المفاهيم أو الأفراد بعضها ببعض. وقد تم تعريف الخصائص التالية:

○ **HasIngredient** : ويشترك منها كل من:

1) **HasBase** (والتي تؤمن علاقات الربط بين مفهومي pizza و pizzaBase) مثال:

Pizza HasBase DeepPanBase أو Pizza HasBase ThinAndCrispyBase أي

البيتزا هي نوعين تختلف حسب نوع العجينة والتي تمثل قاعدة Base البيتزا .

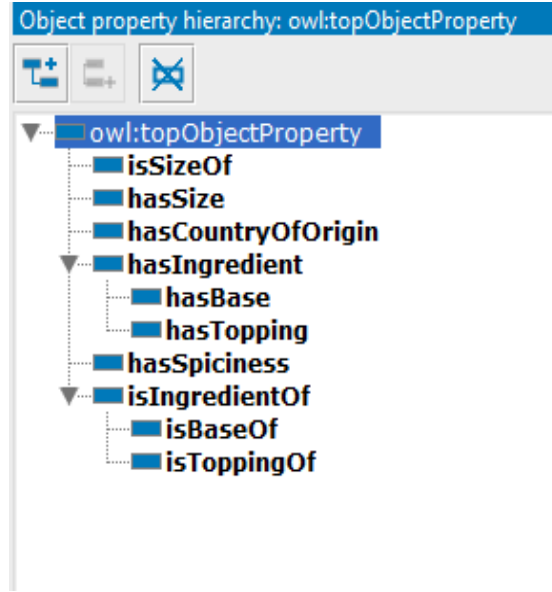
2) **HasTopping** (تؤمن علاقات الربط بين مفهومي Pizza و PizzaTopping) مثال:

CheeseyPizza HasTopping CheeseTopping أي البيتزا من نوع CheeseyPizza

تتصف بوجود الجبنة على سطحها، وكذلك بالنسبة ل MeatyPizza.

○ **HasSize** (والتي تؤمن علاقات الربط بين مفهومي Pizza و PizzaSize) مثال:

Pizza HasSize MidSize أي يكون أحد أحجام البيتزا الحجم المتوسط.



الشكل (15) مخطط يوضح الخصائص في أنطولوجيا النظام

1.3 الخصائص العكسية :

كل خاصية تم تعريفها في الأنطولوجيا المقترحة، لها بالمقابل خاصية عكسية. مثال : "البيتزا تملك عجينة سمكية" تكافئ "العجينة السمكية تمثل قاعدة للبيتزا".

خاصية isBaseOf هي عكس خاصية hasBase، وكذلك hasBase هي عكس خاصية isBaseOf وأيضاً isSizeOf هي عكس خاصية hasSize.

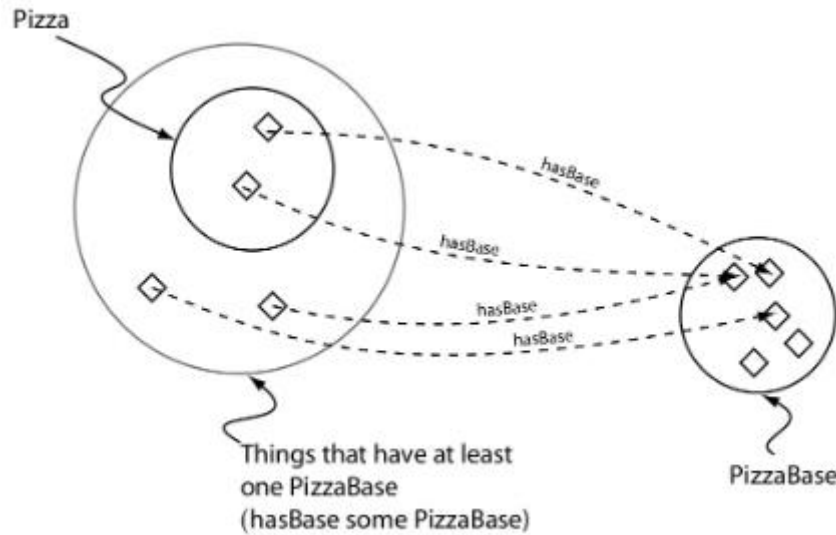
2.3 قيود الخصائص Property Restriction:

تصف هذه القيود مجموعة من الأفراد (Individual) التي تتحدد بصفة ما أو تشترك بصفات محددة. ويمكن تقسيمها إلى ثلاثة مجموعات رئيسية وهي :

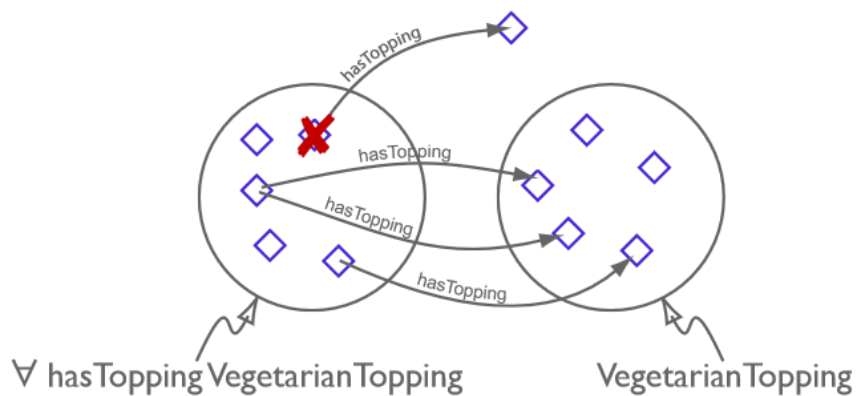
- Quantifier restrictions : وهي (Existential \exists , Universal \forall)
- Cardinality restriction : وتشمل (Min \geq , Equal $=$, Max \leq)
- HasValue restriction : (ϵ)

أشهر أنواع القيود المستخدمة :

- Existential Restriction \exists ويكافئ لفظ some أو يوجد على الأقل ويوضح بالقيود التالي: $\exists \text{ hasBase PizzaBase}$ أي حتى يكون الغرض pizza يجب ان يملك على الأقل نوع من أنواع العجينة PizzaBase.



- Universal Restriction \forall ويكافئ لفظ only ويوضح بالمثال ، من أجل تعريف بيتزا الخضار نضع القيد التالي: $\forall \text{ hasTopping VegetarianTopping}$ أي بيتزا الخضار هي التي تملك topping من نوع خضار فقط .



وبالإضافة إلى استخدام القيود في بناء وتعريف الأنطولوجيا كما لاحظنا في الأمثلة السابقة أيضاً تستخدم في إنشاء استعلامات (Queries) ليتم من خلالها التخاطب مع الأنطولوجيا بهدف الوصول إلى محتوياتها، والحصول على المعلومات المطلوبة الناتجة من العلاقات بين المكونات وذلك حسب القيود المستخدمة في تشكيل الاستعلام ، مثال:

(Pizza and hasTopping some MushroomTopping) وينتج عن تطبيق هذا الاستعلام، جميع أنواع البيتزا التي تملك في إحدى مكوناتها الفطر .

2.5 ربط الأنطولوجيا مع نظام الحوار:

بعد تحقيق عملية البناء، تم الاستفادة من بناء الأنطولوجيا في معالجة بعض الحالات التي تظهر في حوار طلب الطعام، نذكر منها:

• الإجابة عن أسئلة من نمط S-What:

هناك بعض أنواع الأسئلة من نمط S-What ، تم الاعتماد على الأنطولوجيا بهدف الإجابة عليها فمثلاً من أجل:

- " شوفي عندك أنواع بيتزا؟ " يقوم النظام باستخراج كلمتين مفتاحيتين من هذه الجملة وهما "نوع" و"شو" وبالتالي تم تحديد نمط السؤال على أنه استفسار عن أنواع البيتزا المتوفرة ولذلك تم مقابلة هذا السؤال باستفسار Query من الأنطولوجيا وهو ببساطة (NamedPizza) ليقوم بالبحث عن جميع ما يخص هذا الصف من SubClasses, SuperClasses, Instances ومن ثم اختيار الصفوف الفرعية الناتجة منه التي تمثل الإجابة المناسبة.

- " شو في عندك أنواع بيتزا مع فطر؟ " يقوم النظام باستخراج 3 كلمات مفتاحية من هذه الجملة وهي "نوع" و"شو" و"فطر" وبالتالي تم تحديد نمط السؤال على أنه استفسار عن أنواع البيتزا التي يكون أحد مكوناتها الفطر ثم مقابلة هذا السؤال باستعلام Query من الأنطولوجيا وهو (Pizza and hasTopping some MushroomTopping) ليتم البحث عن جميع الأفراد التي تحقق الخاصية المطلوبة.

• الإجابة عن S-Want:

فمثلاً للإجابة على طلب المستخدم "بدي لو سمحت 3 بيتزا الفصول بلا فطر" يقوم النظام باستخراج 5 كلمات مفتاحية من هذه الجملة وهي "بدي"، "3"، "الفصول الأربعة"، "بلا"، "فطر" وبالتالي تم تحديد نمط الجملة على أنها طلب نوع بيتزا معينة مع تعديل على أحد مكوناتها فيقوم النظام من خلال الأنطولوجيا بالتأكد من وجود المكون المطلوب تعديله كأحد مكونات البيتزا الموجودة في الجملة.

5. الفصل الخامس

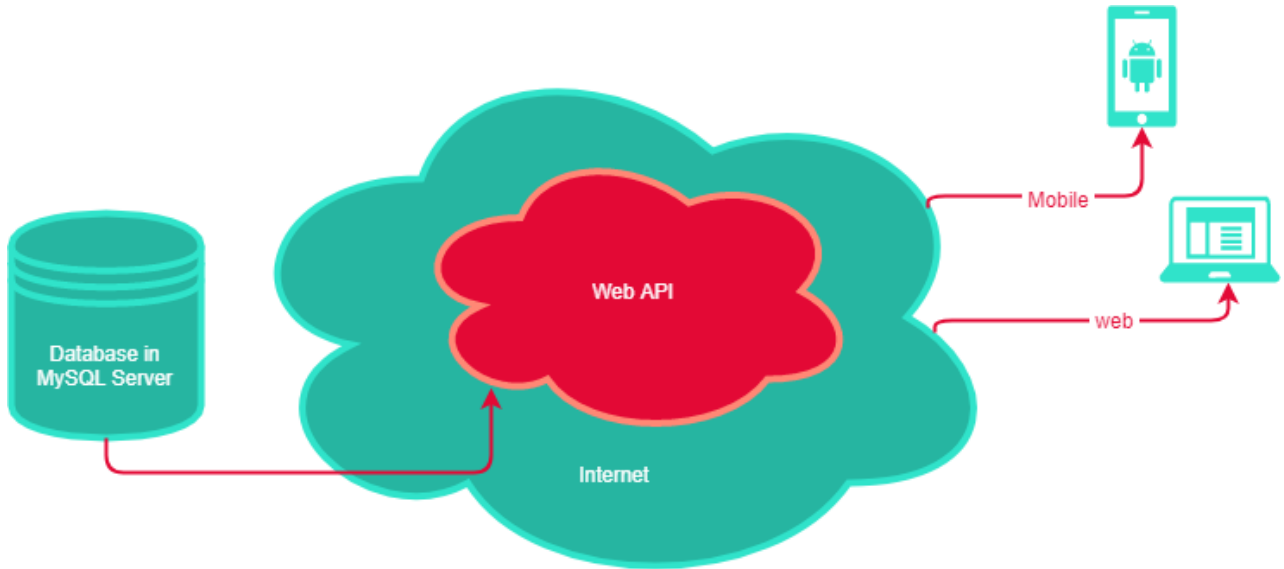
التصميم التفصيلي والتحقيق البرمجي

1. وصف بيئة التنفيذ:

لقد تم التوجه نحو تحقيق المتطلبات الوظيفية الأساسية ضمن وحدة برمجية مستقلة FoodOrdering Web API مما يتيح لأي مطور أن يقوم ببناء تطبيقه من أي منصة (Platform) كانت سواء تطبيق ويب أو تطبيق موبايل، دون الأخذ بالحسبان اللغة التي طُورت بها هذه الـ Web API، وما عليه فقط أن يفهم الدخل والخرج المرجو لكل تابع من التوابع التي تقدمها هذه الـ API للحصول على الاستخدام المناسب وتوظيفه ضمن تطبيقه.

2. تحقيق طبقة الـ WEB API:

في هذه الطبقة أُنجزت كل المهام الأساسية التي يدعمها النظام (المتطلبات الوظيفية). حيث نرى في الشكل (14) كيف أنه بإمكان أي تطبيق أن يتواصل معها عن طريق HTTP Request ويرسل المعلومات اللازمة للمعالجة بصيغة JSON، ويحصل على الخرج الناتج من هذه الـ API أيضاً بصيغة JSON.



الشكل (16) مخطط يوضح تطوير Web API يمكن للتطبيقات أن تتواصل معها عبر الانترنت

وتم اعتماد لغة Java مع بيئة لتطوير الويب على Java 8 تدعى Spark، لتطوير هذه الـ Web API، حيث تقدم بيئة Spark مجموعة من التوابع اللازمة لتحقيق تطبيق ويب بسهولة.

1.2. أهم توابع الـ FOODOREDING WEB API:

- **textProcessing:** وهو التابع المسؤول عن استقبال الرسائل واختبار صحتها قبل إرسالها للمعالجة، وفي حال صحتها يرسل للصفوف المسؤولة عن المعالجة ليستقبل فيما بعد الرسالة المناسبة ويرسلها إلى المستخدم.
- **audioProcessing:** وهو التابع المسؤول عن استقبال الرسائل الصوتية واختبار صحتها قبل إرسالها للمعالجة، وفي حال صحتها وخلوها من الأخطاء يقوم بإرسالها إلى Cloud Speech API ليستقبل رسالة نصية و يرسل للصفوف المسؤولة عن المعالجة ليستقبل فيما بعد الرسالة المناسبة ويرسلها إلى المستخدم.

2.2. أهم الصفوف المستخدمة لتحقيق FOODOREDING WEB API:

الصف API:

ويحوي التابعين الرئيسيين textProcessing و audioProcessing.

الصف Start:

هو الصف المسؤول عن تهيئة المتحولات (DB, ontology, dictionary, ...) وفي حال كانت المحادثة (الطلبية) جديدة يقوم بإنشاء "order_id" ثم يستدعي تابع المعالجة.

الصف Conversation:

وهو الصف المسؤول عن تجزئة الطلب لمجموعات ومعالجته حسب التوابع التالية :

- **Answer:** يدير ترتيب عمليات التواصل بين التوابع.
- **Chunk:** يساعد في عملية named entity.

○ Separate_Orders: يقوم بفصل الطلبات المتعددة في حال تم سرد عدة طلبات في جملة واحدة.

○ Detect_type: يقوم بتحديد نمط الجملة (استفسار عن حجم، استفسار عن نوع، استفسار عن سعر، طلب).

○ Processprice: يحاول الإجابة عن الأسئلة المتعلقة بالسعر.

○ Processdescription: يحاول الإجابة عن الأسئلة المتعلقة بالتوصيف.

○ Processorder: يعالج الطلبية و يقوم باستكمال جميع خصائص الطلب للحصول على الفاتورة النهائية.

○ give_replay: يحدد الجملة التي ستوجه للمستخدم بناءً على القواعد الموجودة.

الصف CreateDB:

إنشاء قواعد المعطيات وتهيئتها للعمل.

الصف DictionaryChunker:

إدارة عمليات الوصول للقاموس، وبناءه اعتماداً على الأنطولوجيا.

الصف MySQLDB:

إدارة عمليات الوصول لقواعد المعطيات والتعديل عليها.

الصف MyString:

تحتوي الجُمْل والإجابات التي يقوم النظام بردها للمستخدم.

الصف Order:

البنية الأساسية للطلب.

الصف Pizza:

الخصائص المتعلقة بطلب البيتزا من حجم وسعر ونوع.

الصف Send:

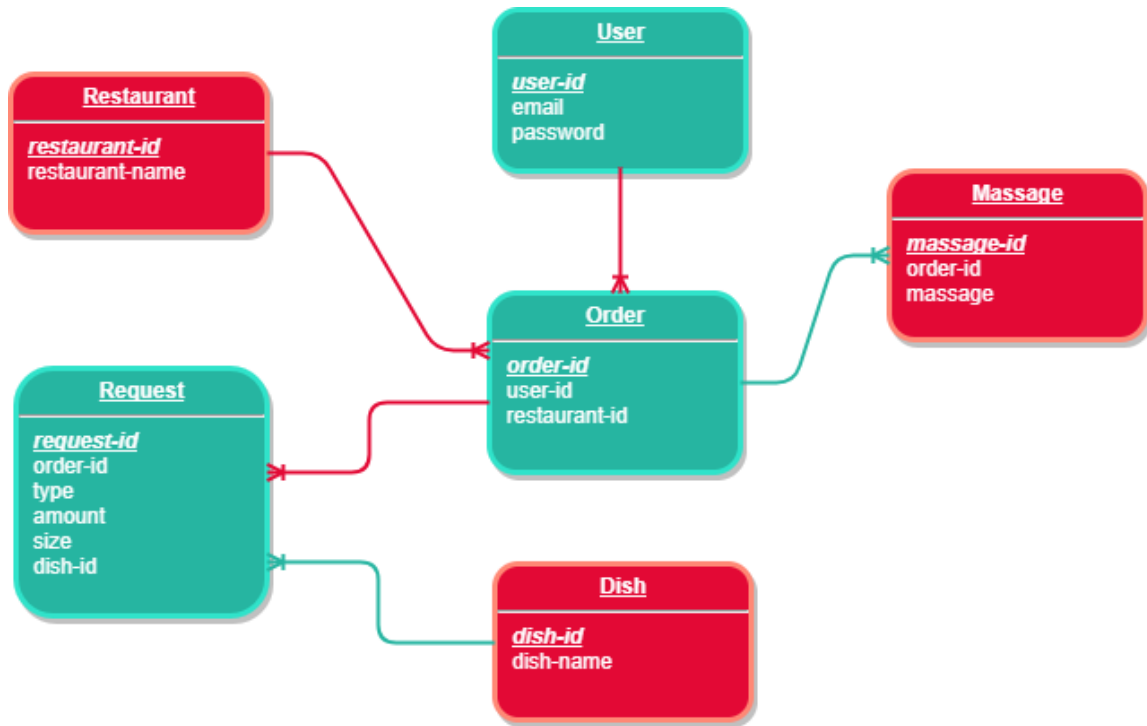
يحتوي تابع تحويل كل request إلى الصيغ المطلوبة (من محارف وأرقام إلى JSON).

الصف LoadingOntologies والصف DLQueryExample :

تحميل الـ ontology وإدارة عمليات الوصول إليها.

الصف VoiceRecognition:

الذي يقوم بالتواصل مع Cloud Speech API وخرج التابع الأساسي فيه هو مجموعة من الـ transcripts المحتملة للصوت الذي تم إرساله، حيث يتم إرسال الصوت كملف صوتي بصيغة .pcm بطريقة Synchronized وذلك من خلال صف (SyncRecognizeResponse) نتيجة هذه الـ API.



الشكل (17) مخطط ERD

3. بناء نظام طلبات مبسّط اعتماداً على الـ FOODORDERING WEB API:

تم بناء نظام بسيط على هذه الخدمة ليوضح كيفية تعامل صاحب المطعم الذي يريد أن يستخدم الـ API ليضيف ميزة الطلب على الانترنت عبر المصادقة إلى نظامه من جهة، وكيفية سير المصادقة بين الزبون النهائي (صاحب الطلب) والوكيل التخاطبي من جهة أخرى.

طُوّر النظام بناء على FoodOrderingAPI باستخدام Android Studio لبناء تطبيق أندرويد، كان عبارة عن عدة activities أهمها:

- Signup Activity: الواجهة التي تسمح للمستخدم النهائي (طالب الطعام) بإنشاء حساب.
- Login Activity: الواجهة التي تسمح للمستخدم النهائي بتسجيل دخول.
- Chat Activity: الواجهة التي تسمح للمستخدم النهائي بإجراء مصادقة لطلب الطعام.
- Address Activity: الواجهة التي تسمح للمستخدم النهائي بتحديد العنوان المراد توصيل الطلبية إليه في حال رغب بخدمة التوصيل، وهي عبارة عن خريطة Google Map التي تقدمها كخدمة لأجهزة الأندرويد، تسمح بتحديد إحداثيات مكان ما على الخريطة ومشاركته من الـ Activities الأخرى لإرساله لاحقاً إلى FoodOrderingAPI من أجل إضافته لعناصر الطلب الرئيسية لإكمال الطلب.

طُوّر الجزء الآخر من النظام والخاص بصاحب المطعم الذي يرغب بإضافة الخدمة إلى نظامه باستخدام PHP CodeIgniter.

4. أهم المكتبات البرمجية والأدوات والمحركات المستخدمة:

- فيما يخص عمليات استخراج الكلمات المفتاحية (Keywords extraction) و كشف أسماء الكيانات
Named entity recognition (NER):
تم الاستعانة بمكتبة **aliasi-lingpipe-4.1.0** التي تم الشرح عنها في الفصل الرابع بالإضافة لعدد من المكتبات المتكاملة معها لإتمام المعالجة.
- فيما يخص الأنطولوجيا وتضمينها:
- **Owl-distribution-3.5.0**: مكتبة تساعد في تحميل الأنطولوجيا المبنية ببرنامج Protege والتي تملك اللاحقة owl، وفي إجراء عمليات الوصول لعناصرها (SubClasses, SuperClasses,...).
- **HermiT**: تساعد على إنشاء وتنفيذ DLQuery من خلال كود الجافا.
- فيما يخص بناء FoodOrdering API
Spark: وهي منصة تطوير للويب على Java 8، باستخدام Eclipse Neon.
- Postman**: أداة لاختبار توابع ال API الت تم تطويرها.
- فيما يخص بناء التطبيق:
Android Studio: لتطوير تطبيق Chat المستخدم النهائي على أجهزة الأندرويد.
- Fast Android Networking**: لإدارة عمليات إرسال البيانات واستقبالها عبر الانترنت.
- PHP CodeIgniter**: لتطوير التطبيق من جهة صاحب المطعم.
- فيما يخص التعرف على الصوت:
Cloud Speech API: من أجل تحويل المقاطع الصوتية التي تتم أخذها من المستخدم إلى نص.
- فيما يخص تحديد عنوان طالب الطعام من أجل التوصيل:
Google Map API: خدمة خرائط تمكن المستخدم من تحديد إحداثيات مكان ما على الخريطة.

5. كيفية تحقيق المتطلبات غير الوظيفية برمجياً:

أثناء مراحل العمل قمنا بمراعاة بعض المتطلبات غير الوظيفية وكان أهمها:

- زمن الاستجابة: والذي كان مقبولاً نسبياً بسبب تكامل عدة عوامل تبدأ من الوقت المعقول لخدمة التعرف على الصوت التي تم استخدامها كمرحلة أولى في التطبيق، ومروراً بعدم وجود خوارزميات معقدة ضمن المعالجة النصية، بالإضافة للسرعة التي تقدمها مكتبات تبادل المعلومات عبر الانترنت في التطبيق النهائي.
- التوافقية مع عدة منصات: إن أهم نقطة تمت مراعاتها أثناء تطوير التطبيق برمجياً هي فصل جزء الكود الخاص بالمعالجة وتغليفه بمجموعة توابع يمكن استدعائها من من أي منصة كانت أو من أي تطبيق بغض النظر عن اللغة التي سوف يُطوّر بها.
- قابلية إعادة الاستخدام: إنّ بنية النظام والمعالجات العامة ضمن الكود والتي تقبل تقريباً معالجة أي نوع من الطعام بأي مطعم بمجرد توافر أسماء أطباق الأطعمة المُقدمة في ذلك المطعم وملحقاتها دون جهد كبير وتعديل مباشر على الكود تضمن هذه الخاصية.
- قابلية إضافة مميزات جديدة في المستقبل: يقبل وبشكل مرّن إضافة أي ميزة جديدة بسبب عزل مهام المعالجة الأساسية في طبقة مستقلة عن الواجهات التخاطبية مع المستخدم.
- سهولة الاستخدام: نضمن تحقيق هذه الميزة في نظامنا وذلك لأنه نظام محادثة بواجهات تخاطبية قريبة من واجهات الدردشة التي يستخدمها الناس يومياً، بالإضافة لعدد أزرار قليل بأيقونات و functionality واضحة.
- الأمان والخصوصية: الحساب الخاص بكل مستخدم وبنية قاعدة المعطيات تضمن نوعاً من الأمان.

6. الفصل السادس

التنفيذ والاختبارات

1. نتائج التنفيذ:

1.1. إضافة صاحب المطعم لأطباقه ومكونات كل طبق لتوسعة الأنتولوجيا بها:



إضافة طبق

بيئزا الفصول الأربعة

--

1500

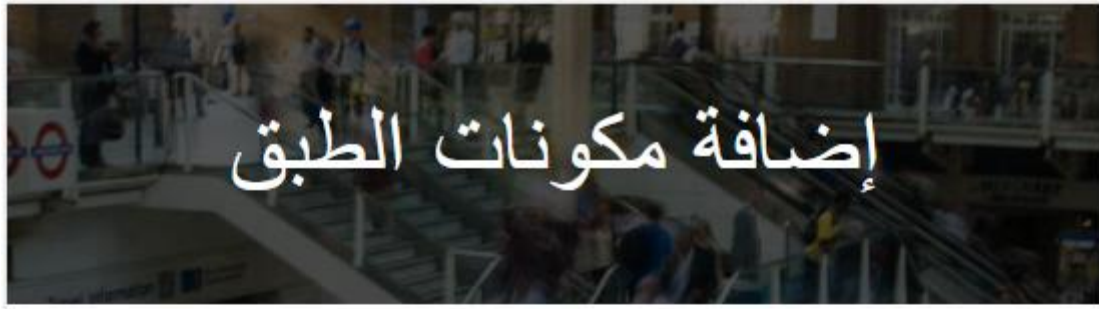
وسط

اختر صورة

إضافة

[العودة إلى الصفحة الرئيسية](#)

الشكل (18) إضافة طبق



<input type="text"/>	بندورة
<input type="text"/>	قليلة
<input type="text"/>	بصل
<input type="text"/>	فطر
<input type="text"/>	نرة
إضافة المكونات	

[العودة إلى الصفحة الرئيسية](#)

الشكل (19) إضافة مكونات الطبق

2.1. المحادثة بين الزبون والوكيل التخابطي:

واجهة تسجيل الدخول :



المثال الأول :



المثال الثاني :



المثال الثالث :



المثال الرابع :



2. المنهجية المقترحة لاختبار النظام ونتائجها:

بسبب صعوبة إيجاد مقاييس ثابتة لتقييم النظم التخاطبية التي يتبادل فيها النظام والمستخدم الحوار والتي تكون غالباً محددة المجال (restricted domain) لأداء مهمة معينة كما في نظامنا، اتجهنا لإجراء تقييم مرحلي للنظام، وفي كل مرحلة قمنا بتطبيق المقاييس المعيارية إن وجدت أو بإجراء اختبارات مباشرة من قبل مستخدمين إن لم توجد.

1.2. اختبار مرحلة تحويل الصوت لنص مكتوب:

في هذه المرحلة التي قمنا فيها باستخدام خدمة من Google لتحويل الصوت لنص، قمنا بإجراء اختبارات مباشرة على مرحلتين:

○ مرحلة التعرف على الكلمات كل كلمة على حدة: حيث تم إدخال حوالي 100 كلمة من الكلمات التي ترتبط بشكل مباشر (كأنواع الأطباق) أو بشكل غير مباشر (ككلمات الطلب أو السؤال: بدي، أدّيش) إلى جزء النظام الخاص بالتعرف على الصوت والمعتمد على Cloud Speech API فكانت النتائج كالتالي:

- عدد الكلمات التي تم التعرف عليها من المحاولة الأولى 64/100.
 - عدد الكلمات التي تم التعرف عليها في المحاولة الثانية حوالي 10/100 مثل (حقّها، سبيسيال، وصّي، سلامي، عيران).
 - عدد الكلمات التي فشل في التعرف عليها 10/100 مثل (بأدّيش، خفّلي، بلا، تنتين ..).
- وتم تحسين دقة التعرف على هذه الكلمات المهمة باستخدام قائمة (Context List) من الكلمات المرتبطة بالسياق يتم تمريرها لتابع التعرف، فأتناء مقارنة الأداة لهذه الكلمات للحصول عليها كنص تقوم الأداة بالبحث عن هذه الكلمات ضمن القائمة المقترحة أولاً مما يضمن دقة عالية.

○ مرحلة التعرف على جُمل الدخل من المستخدم: في هذه المرحلة قمنا بإدخال مجموعة من التسجيلات (على شكل طلبيات) التي كنا قد جمعناها سابقاً لاستخلاص الكلمات المفتاحية المستخدمة في مجال طلب الطعام على الانترنت، هدف هذه المرحلة هو معرفة إلى أي مدى استطاع النظام المبني على Cloud Speech API من Google أن يتعرف على تسجيل يحوي ربما جملة أو أكثر بمعدل 6 كلمات وسطياً في الجملة، وكانت النتائج كالتالي:

- عدد الجُمْل التي تعرف عليها بشكل كامل: 32 من أصل 40 جملة (في حال وجود خطأ في حرف واحد من أحرف إحدى الكلمات كنا نعتبرها صحيحة كلياً).
- عدد الجُمْل التي تعرف عليها بشكل جزئي: 5 جُمْل من أصل 40 حيث كان خطأ التعرف فيها بكلمة كاملة.
- عدد الجُمْل التي فشل في التعرف عليها: 3 من أصل 40 جملة، حيث كان الخطأ في كلمتين كاملتين.

والجدير بالذكر أن الاختبارات في هذه المرحلة أعطت النتائج السابقة لأن مواصفات البيئة المحيطة أثناء تسجيل الصوت هي التي تؤثر على دقة التعرف، لذلك تفاوتت قدرة النظام في هذه المرحلة في التعرف على الكلمات بين محاولة أولى أو ثانية أو حتى ثالثة ورابعة، أيضاً إن Google Cloud Speech API قادرة على التعلم مع الزمن باستخدام البيانات التي يتم إدخالها إليها من قبل المستخدمين أو باستخدام البيانات التي تحصل عليها شركة Google من تطبيقاتها وأنظمتها الأخرى.

2.2. اختبار مرحلة المعالجة النصية:

في هذه المرحلة قمنا بجمع حوالي 100 حوار نصي كامل من الأشخاص في الوسط المحيط، تتضمن هذا الحوارات جُمْل طلب واستفسار وسؤال فيما يخص مجال طلب الطعام على الانترنت. تم إدخال جُمْل الحوار التي حصلنا عليها بشكل فردي إلى جزء النظام الخاص بالمعالجة النصية، لتتم مناقشة أداء النظام في كل جزء منه كالتالي:

مرحلة التقطيع: التقطيع هنا يعتمد على إيجاد الكلمات تبعاً للفراغات فيما بينها، وتقريباً إذا لم يكن هناك خطأ في المرحلة السابقة (التعرف على الصوت) فإن نتائج هذه المرحلة مضمونة تماماً، فأتثناء الاختبارات تم تقطيع جميع الجمل إلى كلماتها بشكل صحيح لأن الخطأ في مرحلة الصوت على مستوى الكلمة الواحدة وليس دمج للمقاطع القريبة من كلمتين متتاليتين، وعليه فإن المحادثات الكتابية من قبل الزبون ممكن أن يحصل فيها خطأ بنسبة أكبر من المحادثة التي يتفاعل فيها المستخدم باستخدام الصوت وذلك بسبب الأخطاء الإملائية المحتملة ونسيان الفراغ بين الكلمات.

مرحلة استخراج الكلمات: أيضاً نلاحظ هنا أن دخل هذه المرحلة يعتمد على سابقتها وعلى عوامل أخرى تتعلق بتنوع الطرق التي يكتب الأشخاص بها الكلمات العامة.

مرحلة اكتشاف الكيانات: أيضاً تعتمد على سابقتها.

مرحلة التجذيع: أفادت في تخفيف الخطأ الناتج عن الأخطاء الإملائية البسيطة وفي حالات استبدال كلمة بجمعها، فتحسنت الدقة.

معرفة نمط الجملة: أظهرت عينات الاختبار أن هناك كلمات معينة تستخدم بكثرة تميز أنماط الجمل.

النتائج النهائية للتقييم:

عدد المحادثات التي استطاع النظام أن يكملها في وقت معقول حوالي 25 محادثة من أصل 40.

عدد المحادثات التي استطاع أن يكملها لكن بتكرار بعض الأسئلة عن المكونات الأساسية للطلب: حوالي 15 محادثات من أصل 40.

عدد المحادثات التي لم يستطع أن يكملها: لا يوجد محادثات لم يفهمها ما يميزه أنه يسأل عن مكونات الطلب حتى وإن ذكرت بصيغة غير مفهومة من قبل المستخدم حتى يحصل عليها.

طريقة أخرى للتقييم :

هناك طريقة إضافية قام بها فريق العمل يدوياً من أجل حساب الدقة الفعلية للنظام، حيث قمنا باختبار التطبيق على 60 حواراً تفاعلياً ومن أجل كل حوار قمنا بحساب مايلي:

○ **Fully**: وهي عدد الأسئلة المفهومة من أول مرة ، والتي يتم إعطاء الإجابة المناسبة عليها فوراً، وفي حال كانت الجملة المدخلة للنظام هي عبارة عن طلب وليس سؤال، أيضاً يتم تسجيله من أول مرة، دون الحاجة للسؤال عنه مرة أخرى.

○ **Partially**: وهي عدد الأسئلة المفهومة من قبل النظام ولكن بعد الحاجة لعدة أجوبة من قبل المستخدم.

○ **None**: عدد الأسئلة الغير مفهومة أبداً بالنسبة للنظام.

الدقة 1 وتمثل النسبة : (عدد الأسئلة Fully + عدد الأسئلة partially) / عدد الاسئلة الكلية في الحوار

الدقة 2 وتمثل النسبة : (عدد الأسئلة Fully) / عدد الاسئلة الكلية في الحوار

والجدول التالي يوضح قيم البنود الخمسة السابقة من أجل كل حوار : وذلك من أجل 20 حوار

None	partially	fully	الدقة 1	الدقة 2
0	0	6	1	1
0	2	1	1	1/3
1	0	3	3/4	3/4
2	0	2	2/4	2/4
1	0	2	2/3	2/3
1	1	3	5/6	3/6
1	0	1	1/2	1/2
0	0	4	1	1
2	1	1	2/4	1/4
0	0	3	1	1
0	1	3	1	3/4
1	0	1	1/2	1/2
0	0	3	3/3	3/3
1	0	1	1/2	1/2
1	0	1	1/2	1/2
0	0	4	4/4	4/4
0	1	2	3/3	2/3
0	1	3	4/4	3/4
0	2	2	4/4	2/4

في حال قمنا بإيجاد النسبة المئوية للعمود الرابع الذي يمثل الدقة 1 نجد الدقة 79.86% للنظام، وبالنسبة للدقة 2 نجد الدقة 58.125%.

فدقة النظام في حال تعرف على الطلب من أول مرة أقل من الدقة في حال طلب النظام إعادة الطلب، أو طلب معلومات تنقصه من الطلب ليكتمل فهم الجملة بعد عدة استفسارات.

مع العلم أن هذه النتيجة تم الحصول عليها من اختبار نماذج حوارات كانت موضوعة على أساس إجراء حوار طبيعي بين أي صاحب مطعم وزبون أي أن البيانات المختبرة لم تخضع لأي قيود ولم تكن على أساس الحوار مع التطبيق.

تحليل نتائج الاختبارات:

إن المرحلة التي توصلنا إليها جيدة في ظل كل تلك المعوقات المتعلقة بمعالجة اللغة العربية وزيادة على ذلك اللهجة العامية التي كما ذكرنا لا تخضع لأية قواعد تسهّل عملية المعالجة أو على الأقل تنظمها وفق مسار محدد يضمن فهم نسبة كبيرة من الحالات مع وجود بعض الحالات الشاذة أو اللبس في أماكن ما ولا تخلو لغة من لبس هنا أو هناك، أما في حالتنا ولعدم وجود تطبيقات أو دراسات مشابهة في نفس المجال للمقارنة معها، فنحسب ذلك إنجازاً بحد ذاته.

آفاق وأعمال مستقبلية:

إن أحد أبرز إيجابيات النظام المقترح هي أنه قابل للتطوير والتوسع بشكل كبير، حيث كانت أحد الخطط المستقبلية لأعضاء الفريق هي مكاملة نظام اقتراح (recommendation System) مع التطبيق، يقوم هذا النظام بمعرفة هوية المستخدم (من اسم وعمر والمهنة و الحالة الاجتماعية والدخل الشهري....) وتفضيلاته بما يخص الطعام والمطاعم بشكل عام، وبالفعل تم تنسيق استبيان يحوي على 24 سؤال تخص النظام المقترح. وتم تعبئة حوالي الـ 1400 استبيان ، الهدف منها معرفة تفضيلات الأشخاص بشكل عام، حيث تم تطبيق أحد طرق تنقيب البيانات (Data Mining) وهي خوارزميات قواعد الترابط (Association Rules). فكانت الخطوة هي تعبئة الاستبيان من قبل كل مستخدم في بداية اشتراكه بالتطبيق لمساعدته لاحقاً في اقتراح وجبات يفترض أن تكون مرغوبة من قبله وذلك بعد دراسة الاستبيان المقدم وحسب البيانات المجمعة سابقاً.

من ناحية أخرى، فإن تجميع أسماء جميع الأطباق المشهورة في المطاعم مع مكوناتها وملحقاتها ليس بالأمر الصعب لكنه يحتاج وقت لبناء وتوسيع الأنولوجيا لتشمل هذه الأطعمة، مما يعني صاحب المطعم من جهد إدخال أسماء الأطباق الخاصة بمطعمه مع مكوناتها، لذلك من الممكن أن نقوم بهذه الخطوة في مراحل لاحقة.

السمات المميزة للـ chatbots الحالية هي قدرتها على التعلم مع الزمن مما يزيد فاعليتها ودكائها، في خطوة لتحقيق هذا الهدف أتحنا تخزين كل المحادثات في قاعدة بيانات النظام، سواء داتا الاختبارات التي نجريها على النظام أثناء تطويره أو داتا يمكن الحصول عليها من مستخدمين فعليين إذا ما تم استثمار التطبيق.

- [1] Ong Sing Goh, Cemal Ardil, Wilson Wong, Chun Che Fung, "A Black-box for Response Quality Evaluation of Conversational Agent Systems", in International Scholarly and Scientific Research & Innovation, 2010.
- [2] Hsien-Chang Wang, Thing-Fa Wangm and Yi-Nan Liu, "A Conversational Agent for Food-ordering Dialog Based on VenusDictatae", in Institute of Information Engineering National Cheng Kung University.
- [3] R.lopez-Cozar, A. J. Rubio, P. Garcia, J. Diaz, "A Knowledge Representation Model for A Voice Dialogue System", in Universidad de Granda, 1997.
- [4] Alexei V. Ivanov, Vikram Ramanarayanan, David Suendermann-Oeft, Melissa Lopez, Keelan Evanini and Jidong Tao, "Automated Speech Recognition Technology for Dialogue Interaction with Non-Native Interlocutors", Educational Testing Service R&D, 2015.
- [5] B.H. Juang# & Lawrence R. Rabiner*, "Automatic Speech Recognition – A Brief History of the Technology Development", in #Georgia Institute of Technology - Atlanta, * Rutgers University and the University of California - Santa Barbara, 2004.
- [6] Anne R. Diekema, Ozgur Yilmazel, and Elizabeth D. Liddy, "Evaluation of Restricted Domain Question-Answering Systems", in Center for Natural Language Processing, School of Information Studies, Syracuse University, 2004.
- [7] Omar Trigui, Lamia Hadrich Belguith, and Paolo Rosso, "DefArabicQA: Arabic Definition Question Answering System", NLP Research Group- MIRACL Laboratory, University of Sfax, Tunisia Natural Language Engineering Lab. - ELiRF, Universidad Politécnica de Valencia, Spain.
- [8] Yassine Benajiba, Paolo Rosso, Abdelouahid Lyhyaoui, "DefArabicImplementation of the ArabiQa Question Answering System's Component"
- [9] B .Hammou, H .Abu-salem, S .Lytinen, M .Evens, "QARAB:A question answering system to support the Arabic language. In the proceedings of the workshop on computational approach to Semitic languages", ACL ,2002
- [10] F. A. Mohammed, K. Nasser, H. M. Harb, "A knowledge based Arabic question answering system (AQAS)." ACM SIGART Bulletin, 1993.
- [11] Brini, Ellouze, Mesfar, Hadrich Belguith, QASAL :“An Arabic question-answering system for factoid questions”IEEE Xplore Conference : Natural Language Processing and Knowledge Engineering, 2009.
- [12] David Milward & Martin Beveridge, “ Ontology-Based Dialogue Systems” Advanced Computation Laboratory, Cancer Research UK 44 Lincoln’s Inn Fields, London, UK.

[13] Holge Knublauch & Martin Beveridge, “ Ontology-Based Dialogue Systems” Advanced Computation Laboratory, Cancer Research UK 44 Lincoln’s Inn Fields, London, UK.

– LingPipe :<http://alias-i.com/lingpipe/demos/tutorial/ne/read-me.html>

–**Google Speech API**: <https://cloud.google.com/speech/docs/>