



BOTNATION.AI



All you need to know about Artificial Intelligence by Botnation

Contents

Introduction	3
Before getting started ...	4
1.1 Do you really need an AI?	4
1.2 The limits of AI	5
1.3 The Default Answer sequence	5
1.4 What type of Artificial Intelligence can we find on Botnation?	6
1.5 Is it complicated to configure?	7
1.6 What degree of complexity is best suited?	7
1.7 Why do we have to start from scratch?	7
1.8 The more the merrier	8
Basic Features	9
2.1 Keywords and Expressions	9
2.2 Negative Expressions	13
2.3 Priority Keywords (or Exit Keywords)	15
Advanced Features	17
3.1 Contexts	17
3.2 Keywords Libraries	18
3.3 Tunnels	19
3.4 How do I find which rules are in conflict?	24
After launching your chatbot	25
4.1 What to do when users ask multiple questions at once?	25
4.2 Autonomous learning	25
4.3 A chatbot's AI is never over	26
4.4 Recommendation Engine	26
NLP V2 : Next Generation	28
5.1 More than an evolution ...	28
5.2 Keywords and Expressions V2	29
5.3 Recommendation Engine V2	30
5.4 Keyword Libraries V2	31
Conclusion	32

Introduction

“Artificial Intelligence”, this simple association of words makes you dream or scares. **AI is taking more and more place in our daily lives** and, in recent times, seems to be the solution to all problems big and small.

And when it comes to **automated conversational experiences**, we think, logically, that this is an area where AI is going to work wonders.

Botnation, as a market leader, has **innovative technology that will allow your ChatBot to learn to respond on its own**, but you might think that this technology can be extremely complex to implement.

This is not the case, **on Botnation it is not complicated. Our approach is quite simply different**. When other technologies require the intervention of a specialist, we have chosen a **“no-code” and ergonomic interface** which is **directly aimed at content creators** (Digital Agency Project Manager, Marketing Department, Communication Department, Client, Community Manager, etc.).

Our AI technology is very advanced, but we have concentrated our efforts to keep it **simple to use**. We often present it as **“Simple... but powerful!”**

In addition, **we are not dependent on any external AI technology** (Google, Amazon, Microsoft, etc.). Your data will be processed only by Botnation **proprietary algorithms fully developed in France**.

Our tools will allow you to teach your ChatBot how to respond. **You remain in control of his apprenticeship**.

In this guide, **we hope to answer all the questions you have on this topic** and even those you haven't thought of. All this is **complemented by many practical examples** and references to articles in our Online Help to find out more.

Before getting started ...

1.1 Do you really need an AI?

Do not respond too quickly because it is a choice that is not without consequences:

- it takes a long time to implement which implies a **delay in putting online** and **increases the cost of setup**,
- this requires **post-launch monitoring** and maintenance and therefore an **additional regular cost**.

Developing the AI of a chatbot takes time and imposes a particular logic. So, before you dive into the wonderful world of Botnation AI, ask yourself this simple question: "Does my chatbot really need it?"

You can very well build an efficient chatbot that the user interacts with only through buttons. This is quite sufficient in 70% of cases. Even for the most complex content (like FAQs) where **you can guide the user according to the funnel principle to a very precise answer**.

You can also **launch a first version of your chatbot without AI** and configure it later. This will reduce development time and allow you to confront your chatbot with its users more quickly.

If you choose not to use AI, you must disable user input in the "Settings" of the chatbot under the "Chatbot configuration" tab.

In the case of a Web-Bot (chatbot for your website), you can also limit the AI to part of the chatbot by activating or deactivating user input according to the displayed sequence.

Online help:

[Enable / Disable user input on a sequence in my web chatbot](#)

[How to disable the chatbot keyboard](#)

1.2 The limits of AI

For starters, **no business needs a chatbot that has all the answers.**

Which is a good thing. Because as well configured as possible, even with the time and resources involved, **your AI will never be able to answer all of your users' questions.** And this is normal. It suffices to note that even Apple, Amazon and Google have not yet succeeded, although there are decades of research behind their Artificial Intelligence.

The important thing is to **stay in your area of expertise / profession.** It is easier to develop an AI in a particular domain with its uses, its vocabulary and customers / prospects whose behaviors we know.

Plus, **your users don't expect your chatbot to have everything answered.**

But it also means that we absolutely have to deal with this situation.

And how you do it matters a lot in how users perceive your chatbot.

In these cases, the **Default Answer sequence** is triggered automatically. It is therefore necessary to configure it properly.

1.3 The Default Answer sequence

Definition :

This is the sequence that is triggered when your chatbot's AI cannot find an answer.

This is a passage that must absolutely be taken into account (see previous paragraph) and your chatbot will be judged according to how you managed it. **This is, for us, the most important part of your chatbot!**

Do not repeat yourself

Nothing is more frustrating than having a chatbot repeating the same phrase over and over: "Sorry, I don't understand"

The minimum is to allow variations in the responses and Botnation allows you to do this with its native A / B Testing features.

Exemple: Rotation of several texts: "I'm not sure I understand", "I'm just a ChatBot, I can't know everything", etc.

Do not leave the user unanswered and unassisted

It is one thing to say that we didn't understand, but helping the user to use the chatbot is much better. It is therefore necessary to provide solutions. These can take three forms:

- rewording,
- reframe the field of expertise,
- contact a human.

For example, you can configure your ChatBot so that **on the 1st pass, it offers the user to formulate their request using a different vocabulary.**

If the chatbot still does not understand, **on the 2nd pass, the chatbot can re-explain its scope of action** and offer the topics covered in the form of buttons and / or FAQs.

And if there is a **third passage** through the default sequence, it is surely that the user's questioning is outside the competence of the chatbot and that a human intervenor is necessary. **The chatbot will therefore offer a way to contact a human** either by phone, by leaving a written message via a form integrated into the chat (or dialogue) interface or passing the baton to a human agent for a live chat session in the chatbot interface. On Botnation, this feature is called "**human escalation**".

Online help:

[Transfer function - escalation - to transfer the discussion to a human](#)

1.4 What type of Artificial Intelligence can we find on Botnation?

We use **Natural Language Processing (NLP)**. It is a field involving **linguistics, computing and artificial intelligence**, which aims to create natural language processing tools for various applications.

It is on this discipline of Artificial Intelligence that Botnation focuses its research and develops its own algorithms.

In the area of conversational experiences that interests us, it is about **analyzing** what the user of a chatbot enters, **understanding** it and **reacting** accordingly.

1.5 Is it complicated to configure?

We've done everything to make your job easier. Our user-friendly interface may even seem simplistic to you. But make no mistake, behind the facade there are over a dozen proprietary algorithms working for you.

Simple and efficient dialogue management can be set up in a matter of days just by using basic NLP tools.

And **for a deeper AI, our powerful advanced features** (which we cover below) **are there to cover all your needs** without sacrificing ease of use.

1.6 What degree of complexity is best suited?

The number of sequences is a good indicator to determine the complexity of your chatbot. So the more sequences your chatbot has, the more you'll need to use Botnation's advanced NLP tools.

Suggested scale:

- 20 séquences = Elementary Functions are sufficient
- +20 séquences = Advanced features recommended
- +30 séquences = Advanced Features required

A good practice is to mix a funnel navigation by following a classic tree structure of hierarchy of information **with a minimum of NLP**. You can focus your NLP on detecting major topics and display a classic menu, with buttons, that deals with the selected topic.

After launching your chatbot, you can regularly analyze the issues that have not been understood (see paragraph on our Recommendation Engine) and gradually develop your NLP accordingly.

1.7 Why do we have to start from scratch?

The question indeed arises: why does Botnation not have a pre-configured NLP that would serve as a basis for you?

NLP rules depend on two things: chatbot structure and vocabulary.

As far as the structure is concerned, **each chatbot being different, we cannot impose a predefined shape on you when you need a tailor-made one.**

In addition, the vocabulary used by the users of **your chatbot is specific to your activity.** However, our platform is generalist and does not offer a sectoral approach, because it is not specialized for a particular trade. **Botnation is a tool that hundreds of different professions use.**

Certain expressions have a different meaning depending on the profession, the context or even the region.

And even if for some expressions the meaning is the same for everyone, the way in which we answer them may be different depending on the chatbot.

On the other hand, our chatbot templates / models still have a pre-configured NLP that you can modify to make it more suited to your needs.

And if you develop several chatbots in the same vertical, you can share part of the NLP by using the **Libraries** which we will talk about later.

Finally, when you duplicate a chatbot, the AI rules are also duplicated. So if you use one of your chatbots as the basis for another, you won't have to redo all the AI.

Online Help:

[Chatbot Templates](#)

[Use a template to create a chatbot](#)

1.8 The more the merrier

Botnation allows you to invite your teammates to join you on the development of the chatbot. You may share the tasks, especially in the testing phase. **Everyone will have their own unique way of interacting with the chatbot and this will give you ideas for AI rules to complete or add.**

Basic Features

2.1 Keywords and Expressions

To properly configure your AI, you have to understand how a chatbot analyzes what users enter. We based our NLP on the detection of isolated or associated keywords.

Definition of an Intent:

In the context of Artificial Intelligence, Intent refers to the goal the user has in mind when entering text.

Example: *If the user enters "I am looking to own a mansion" their Intent is "buying a house".*

Definition of a Keyword:

This is a word that triggers a response from the chatbot. You have to select words that will be most frequently used by users to express an intent. You have to think about synonyms and sometimes conjugations.

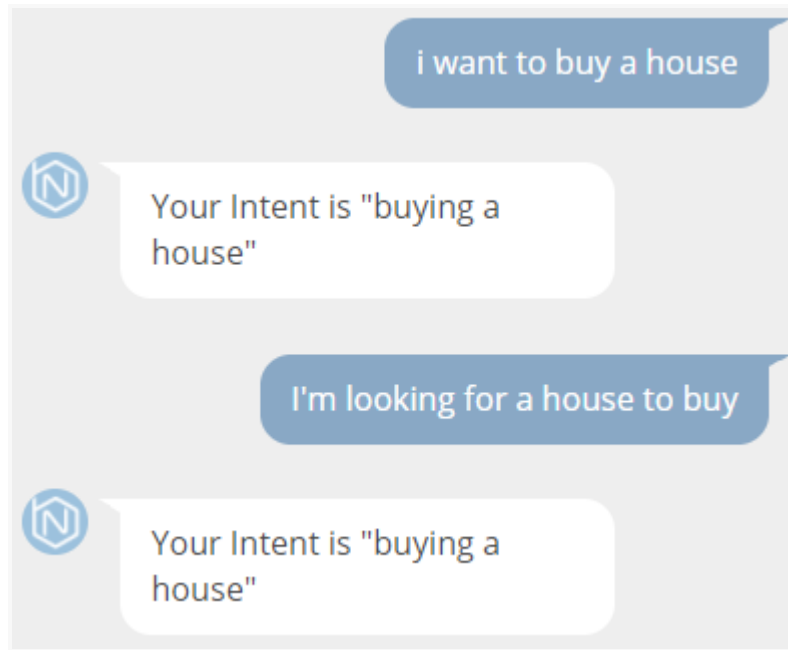
Example : "buy", "purchase", "acquire"

Definition of an Expression:

It is a combination of keywords that usually distinguish two similar Intents. When using an Expression in an AI rule, the rule is triggered only if all the keywords for the expression are present. However, the order in which these words were entered by the user or if they are separated by other words does not influence the triggering of the rule.

Example: *The Expression "buy house" triggers a response even if the user typed "I am looking for a house to buy."*





Definition of an AI rule:

It is a set of Keywords and Expressions corresponding to an Intent that trigger a response from the chatbot.

The chatbot does not understand a whole sentence, but it will recognize certain words. It is by comparing these words with those that have been configured in the rules of his AI that he will be able to associate them with the correct answer.

The selection of the right Keywords and Expressions is therefore essential for an effective AI.

You have to look for the differentiators between the rules and translate the Intents you want your chatbot to understand into relevant Keywords.

Exemple: *Imagine a chatbot specializing in the sale (and only the sale) of real estate.*

We want the chatbot to react to the desire to buy a house, which is differentiated from the desire to buy an apartment. There are two Intents: the type of transaction (purchase) and the type of property (house).

As this is a sales-centric chatbot, there is no need to detect the type of transaction. It is the type of property that is the differentiator.

So all you have to do is make rules about synonyms for house and apartment.

"Buying House" Rule:

house × mansion × homestead ×

"Buying Apartment" Rule

apartment x condo x flat x

You must always enter Keywords with the correct spelling. Botnation algorithms will handle most typos or plural errors. However, when the user makes too many mistakes in a single word, the chatbot can sometimes fail to understand and refer to the Default Answer Sequence. The misunderstood words will then appear in the Recommendation Engine. Only these errors should be added to your AI rules. We will deal with this point later.

Example: *the keyword "purchase" will be recognized even when it is written:*

- *purrchase*
- *purchases*
- *murchase*

But it will not be understood if it is written:

- *purrshase*
- *porchase*

Also **don't worry about typographical cases**, whether your keywords and phrases are in upper or lower case the algorithm will treat them the same.

Use an Expression only if one of the words is common to another rule.

Be careful, the AI only triggers a rule if and only if all the words of an Expression are contained in the user's sentence. So **the more words there are in the Expression, the less likely the user is to use that combination.**

Expressions of more than 2 words must be limited to strict necessity.

Do not include pronouns, conjunctions, prepositions, definite articles or contractions: the, it, you, and, or, if, etc.

Example: *Now imagine our real estate chatbot also wants to offer the sale of property. The chatbot must react to the intent to buy or sell a house or an apartment. That makes four possibilities so you have to create four AI rules. And when we think about it in keywords, we realize that there are going to be words common to several rules. We will not be able to use isolated keywords, we must necessarily make associations to differentiate the intentions.*

First, you have to imagine the different ways in which users might express purchase intention:

- *I want to buy/purchase ...*
- *I am considering purchasing*
- *I would like to own ...*

Et pour l'intention de vente on a :

- I want to sell ...
- I am considering the sale of ...
- I sell...

When we isolate the differentiating elements, this gives:

- for purchase: buy, purchase, purchasing, own
- for sale: sell, sale.

For the type of property we have:

- synonyms for house: house, mansion, homestead
- synonyms of apartment: apartment, condo, flat.

Which gives the following rules:

"Buying House" Rule

buy house ✕ buy mansion ✕ buy homestead ✕ purchase house ✕ purchase mansion ✕
 purchase homestead ✕ purchasing house ✕ purchasing mansion ✕
 purchasing homestead ✕ own house ✕ own mansion ✕ own homestead ✕

"Buying Apartment" Rule

buy apartment ✕ buy condo ✕ but flat ✕ purchase apartment ✕ purchase condo ✕
 purchase flat ✕ purchasing apartment ✕ purchasing condo ✕ purchasing flat ✕
 own apartment ✕ own condo ✕ own flat ✕

"Selling House" Rule

sell house ✕ sell mansion ✕ sell homestead ✕ sale house ✕ sale mansion ✕
 sale homestead ✕

"Selling Apartment" Rule

sell apartment ✕ sell condo ✕ sell flat ✕ sale apartment ✕ sale condo ✕ sale flat ✕

If in the same rule you have an isolated Keyword that is found in an Expression, this Expression is useless because the rule is triggered anyway as soon as this keyword appears in the sentence..

Example:

If we have the following rule:

"Bank Account Problems" Rule

bank account ✕ banking ban ✕ overdraft ✕ bank ✕ banking ✕ agio ✕
banking problem ✕

Since we have "**banking**" and "**bank**" as single keywords, the rule is triggered when the user enters any sentence containing those words. This includes the sentences containing "banking problem", "bank account" and "banking ban" and therefore makes these Expressions unnecessary.

In the end we have:

"Bank Account Problems" Rule

overdraft ✕ bank ✕ banking ✕ agio ✕

Between two conflicting rules, the AI will choose the one that has been used the most statistically and not the one that is potentially the most relevant (see the paragraph on Machine Learning).

Optimizing AI also means using as few rules as possible with as few Keywords and Phrases as possible. The fewer there are, the more responsive the chatbot will be in finding the right answer.

Online help:

[Make Your Chatbot Even Smarter with NLP](#)

2.2 Negative Expressions

Definition:

This is an Expression that excludes a keyword. Just add the - (minus) sign in front of the keyword you want to exclude.

Example: "house -brighton" is triggered when the sentence entered **contains** the Keyword "house" **AND does not contain** the word "brighton"

This is a useful feature when you want to distinguish two rules that might overlap.

Example: In our real estate chatbot, we want to distinguish between requests for

houses with a swimming pool and those without a swimming pool.

We therefore make two rules:

“House without swimming pool” Rule

house ✕

“House with swimming pool” Rule

house pool ✕

The problem here is that “house” is common to both rules. The AI won’t know what to choose between the two rules at first. Then with our learning algorithms, the AI will choose the most popular answer of the two (see paragraph on Machine Learning). But then we have a one in two chance of giving the wrong answer.

One of the ways to solve this problem is to use a Negative Expression excluding “pool” from the first rule.

“House without swimming pool” Rule

house -pool ✕

“House with swimming pool” Rule

house pool ✕

When using a Negative Expression, it must be active on all the keywords and phrases in a rule.

Example: For our rule about houses without a pool:

“House without swimming pool” Rule

house -pool ✕ mansion -pool ✕ homestead -pool ✕

We see here one of the limits of Negative Expressions because if we have two rules with several keywords, we must make all the combinations. In addition, our algorithms will not apply to negative keywords (for reasons of response time optimization) and will not detect input errors on them.

Example: For our rule about houses with and without a pool:

“House without swimming pool” Rule

house -pool ✕ mansion -pool ✕ homestead -pool ✕ house -basin ✕ mansion - basin ✕
homestead -basin ✕

“House with swimming pool” Rule

house pool ✕ mansion pool ✕ homestead pool ✕ house basin ✕ mansion basin ✕
homestead basin ✕

We will see later a more efficient way on Botnation to resolve this kind of conflict of rules with the Priority Keywords.

We will therefore reserve Negative Expressions for chatbots whose AI is not very complex.

2.3 Priority Keywords (or Exit Keywords)

Definition :

It is a Keyword that will trigger a rule, even if the phrase the user enters contains other keywords (or Expression) that could have triggered other rules. To activate this option on a Keyword, it is preceded by an _ (underscore).

Example: If we take the example above and want a rule to be triggered when the Key Word “House” is used alone in a sentence but another rule is triggered when the user enters “I’m looking for a house with a pool”.

We therefore have the same problem as before but by putting “pool” in Priority Key Word, the solution is really much simpler.

“House without swimming pool” Rule

house ✕ mansion ✕ homestead ✕

“House with swimming pool” Rule

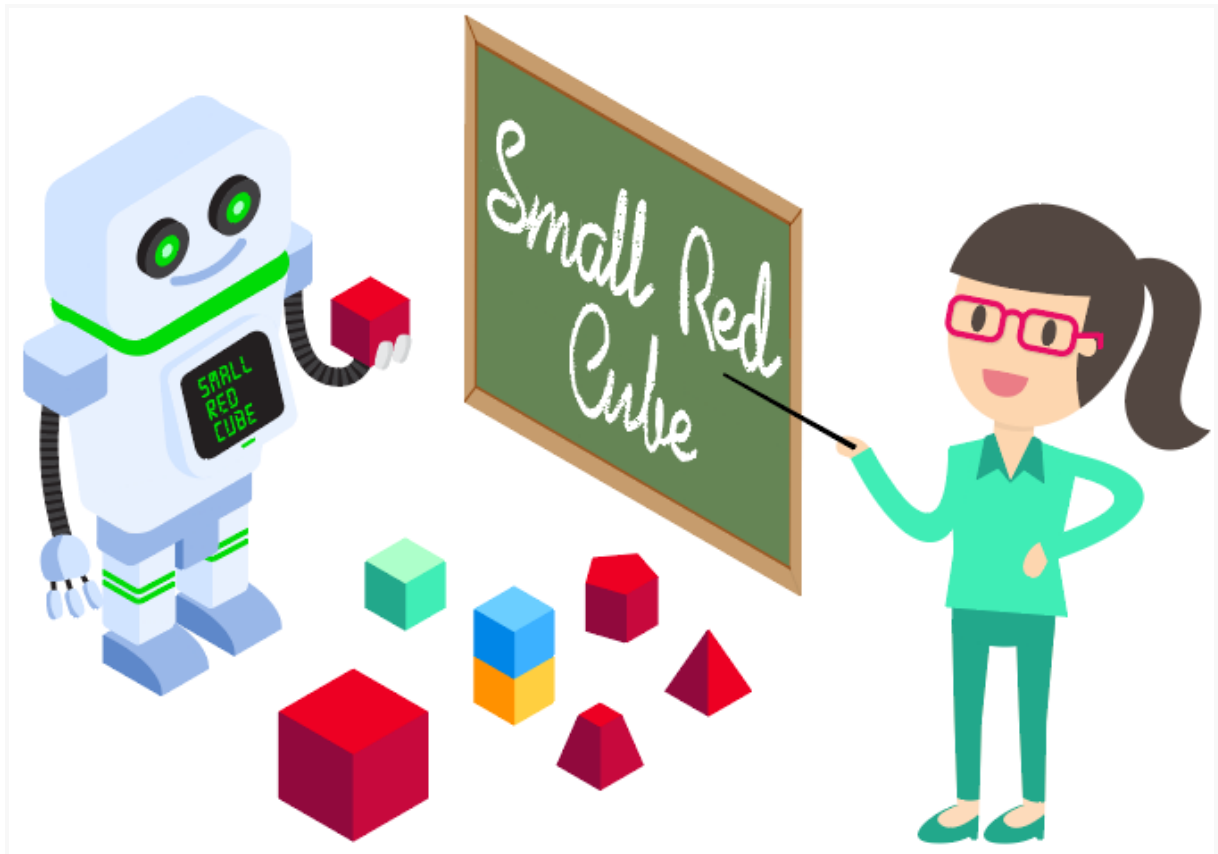
_pool ✕ _basin ✕

The Priority Keywords unfortunately have their limit because **they do not work on**

Expressions. And in a complex chatbot, rather than creating a complicated set of rules, we prefer to use the advanced AI features of Botnation, i.e. Contexts and Tunnels.

[Online help:](#)

[Exit Keywords in NLP](#)



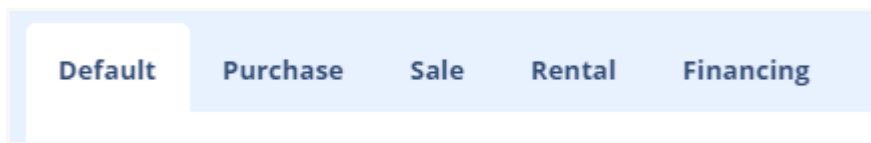
Advanced Features

3.1 Contexts

Definition :

The dictionary definition gives “Set of circumstances in which an event occurs”. In the context of our AI, the meaning is the same and in practice it is a subset of AI rules that we associate with part of our chatbot. Your ChatBot will therefore have different AIs depending on the context of the discussion. It's also a good way to organize the rules by topic.

Example: *In the case of our Real Estate ChatBot, we could have specific contexts for “Purchase”, “Sale”, “Rental” and “Financing”.*



To activate Contexts on your chatbot, you have to do it in the advanced settings.

The main context is the “default context”. We call “Secondary Contexts” the additional contexts that you create.

The default context must always remain the main context of the chatbot.

For a Secondary Context to be taken into account / saved, it must contain at least one AI rule.

A Context is associated with a sequence in the parameters of the sequence (access by the gear icon on the far right in the title bar of the sequence in question).

The rules of a context apply as soon as a sequence to which it is associated is triggered.

In a context, if no default sequence is defined, if the AI does not find a matching rule in this context, it launches a search in the default context. And if there is still no matching rule in the default context, the default response from the default context is displayed.

If a default sequence has been defined in the secondary context:

Search in the AI of the secondary context -> if no response -> Default response from the secondary context.

If a default sequence has not been defined in the secondary context:

Search in the AI of the secondary context -> if no response -> Search in the AI of the default context -> if no response -> Default response of the default context.

***TIPS:** Priority keywords are very useful for getting out of a Context.*

Before discovering the Tunnels, we will focus on a very useful feature in the management of keywords: the Keyword Libraries.

Aide en ligne :

[Introduction to the Context](#)

3.2 Keywords Librairies

Definition :

These are sets of words that you can save and use as keywords in multiple chatbots and / or in different contexts within the same bot.

Library management is done in the AI tab of your chatbot.

To create one, make a list of Keywords or Expressions as if it were a regular AI rule but without Priority Keywords or Negative Phrases. You can then add it to any AI rule in your bots.

***TIPS:** the modifications are taken into account wherever the Library is used.*

This feature is particularly **useful when managing different bots that have the same theme.**

Exemple: *You are a digital agency specializing in real estate.
You could create the following Libraries:*

house

Keywords and key phrases

house × homestead × mansion × home × manor × cottage × chalet ×

Add keywords and key phrases then click enter

apartment

Keywords and key phrases

apartment × condo × flat × lift × studio × Add keywords and key phrases then click enter

And you will be able to use these Libraries in any bots that need to identify these types of assets.

And it also works from channel to channel. You can use your libraries on your Web chatbots and Messenger interchangeably.

Obviously, you cannot use the same Library several times within the same context.

Online help:
[NLP's keywords libraries](#)

3.3 Tunnels

Definition of a Tunnel:

This is a priority AI rule that triggers a search in another context. Optionally, an intermediate sequence can be started.

This intermediate sequence can be used, among other things, to store data in passing in a variable without displaying anything in the chatbot.

Definition of a Variable :

This is data that the chatbot will store in its memory. A variable is defined by its name and by its value.

Example: *The Variable with the name “property_type” can take the value*

“House” or “Apartment”

Online help:

[Get To Know Everything There Is About Variables {{...}}](#)

It is a complex logic to master, but it is a very powerful tool. It will indeed make it possible to detect several intents in the same sentence. The Tunnel in the default context detects a first Intent and then searches the AI rules of another context for the second Intent.

It is also possible to put Tunnels in the secondary context and therefore to detect a third Intent. In theory, using Cascading Tunnels you can detect as many intentions as you want in the same sentence. But from a practical point of view, it is better to limit yourself to three intentions.

The best way to understand how it works is to study a practical case.

Example:

Also in our London real estate chatbot, we want to detect, in natural language, the type of transaction, the type of property and the district concerned in order to be able to launch a search in an ad database.

It is therefore necessary, not only for the chatbot to understand a sentence such as “I am looking for an apartment for sale in Camden” but also to translate this information into data that can be understood by the search engine.

Cascading tunnels and their intermediate sequences will be able to solve this problem.

The general idea is to detect each intent one after the other.

We will need 3 contexts: the default context in which we will detect the “Transaction type” intention, a “Asset type” context to detect this intention and finally a “District” Context for this intention.

Default context

We create a first Tunnel “Purchase” with the keywords corresponding to this intention and which will launch a search in the Context “Type of asset”.


Tunnel : Purchase
Keywords and key phrases

buy ×

purchase ×


own ×

Add keywords and key phrases then click enter


+ 

Add a list of keywords

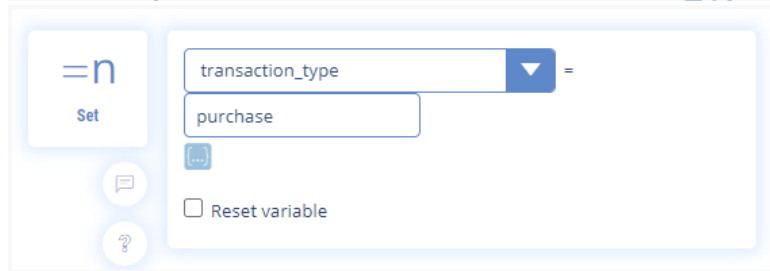
Triggers a search in the context :

Type of asset ▼ 

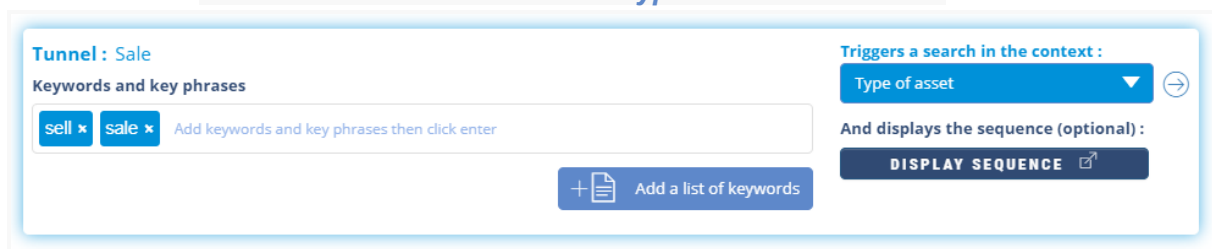
And displays the sequence (optional) :

DISPLAY SEQUENCE 

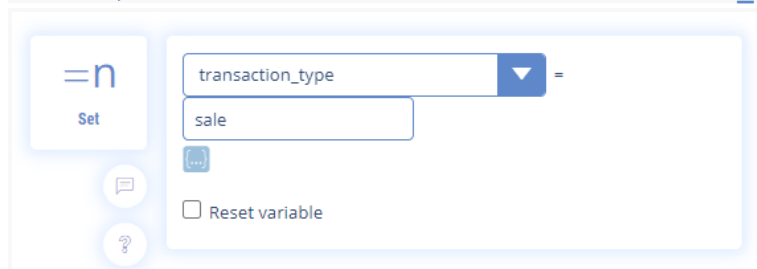
And we use the intermediate sequence to store the information by assigning the value **"purchase"** to the variable **"transaction_type"**.



We then create a **"Sale" Tunnel** with the corresponding keywords and which will also launch a search in the **"Type of asset" Context**.



And we use the intermediate sequence to store the information by assigning, this time, the value **"sale"** to the variable **"transaction_type"**.



We can therefore see that we can have 2 different tunnels which refer to the same context, but since their intermediate sequences and their keywords are different, they detect two different intentions.

The chatbot can now therefore make the difference between "I am looking to **sell** my apartment in Camden" and "I want to **buy** an apartment in camden".

Context "Type of asset"

We create a **"Apartment" Tunnel** with the corresponding keywords and which will launch a search in the **"District" Context**.

Tunnel : Apartment

Keywords and key phrases

apartment * condo * flat * Add keywords and key phrases then click enter

+ Add a list of keywords

Triggers a search in the context :
District

And displays the sequence (optional) :
DISPLAY SEQUENCE

And we use the intermediate sequence to store the information by assigning the value "apartment" to the variable "asset_type".

=n
Set

asset_type =
apartment

☐ Reset variable

We then create a "House" Tunnel with the corresponding keywords and which will also launch a search in the "District" Context.

Tunnel : House

Keywords and key phrases

house * mansion * homestead * Add keywords and key phrases then click enter

+ Add a list of keywords

Triggers a search in the context :
District

And displays the sequence (optional) :
DISPLAY SEQUENCE

And we use the intermediate sequence to store the information by assigning, this time, the value "house" to the variable "asset_type".

=n
Set

asset_type =
house

☐ Reset variable

The chatbot can now make the difference between "I'm looking to sell my **apartment** in Camden" and "I'm looking for a **house** for sale in Camden"

Context "District"

In this context, we create simple AI rules to detect each district, each of which will refer to sequences in which we will **store the name of the district in the variable "district"**.

Rule : Camden

Keywords and key phrases

camden x Add keywords and key phrases then click enter

+ Add a librarie

+ Add a list of keywords

Triggers the sequence :

Camden

Camden Keywords 1

Set

district = camden

☐ Reset variable

To summarize:


In the default context, the chatbot detects the type of transaction then searches for the type of asset in the "Type of asset" context and finally detects the district in the "District" context.

And at each step, the chatbot stores the corresponding info in variables.

*So if the user writes "**I'm looking to buy an apartment in Camden**" the chatbot will detect the intentions and translate them as follows:*

- *transaction_type = **purchase***
- *asset_type = **apartment***
- *district = **camden***

I'm looking to buy an apartment in Camden

 **Data captured by the chatbot:**

Transaction: purchase
Asset: apartment
District: camden

The chatbot can then launch the search in the database because this is information that can be used by the search engine.

Detecting multiple intents in a typed sentence is a headache for all AIs around the world. With tunnels, our platform finally offers a pragmatic and effective solution to this problem.

3.4 How do I find which rules are in conflict?

The more advanced an AI, the longer the setup time and the more likely it is to have conflicting rules. Indeed, **as soon as we add a new element** (keyword, rule, context, tunnels), **we must potentially manage all the conflicts that this generates in cascade.**

To help you resolve conflicts between several rules, we still have a handy tool that are context and expression variables:

- **{{BN_ACTIVE_EXPRESSION}}** which displays the last Expression (or keyword) that was triggered
- **{{BN_ACTIVE_CONTEXT}}** which displays the last Context that was triggered

When you identify a problem in your AI it is often because the bot gave the wrong answer. You will have a possible solution by adding a "text" element with these two variables in the sequence of the wrong answer and replicating the error. **This will tell you which Key Word or Phrase triggered this response as well as the Context in which the mistake was made.**

When you start to use tunnels it may happen that the ChatBot gets stuck in an infinite loop (Ex: In Context A, a Tunnel launches a search in Context B and finds another tunnel that links to the first context.)

In this case, **our platform detects the problem, stops the ChatBot and displays, in the conversation, the Sequence and the Context that triggered the infinite loop.**

Building deep AI is iterative work and there is no other solution than to spend a lot of time on it. It can range from a few days to several weeks depending on the complexity.

This information should therefore be taken into account when drawing up your specifications and planning.

After launching your chatbot

4.1 What to do when users ask multiple questions at once?

When a user asks several questions at once or enters a long text, it is usually to approach complex topics with subtleties that a simple sentence could not sum up. In this case, the chatbot will only react to a few keywords and there is therefore very little chance that the AI will give a relevant response. **It is better to turn off the AI for this specific case** and offer the user other means of obtaining a response such as putting them in contact with a human operator via live chat, by phone or by leaving a message. **This temporary deactivation will only be triggered in this case and will reactivate immediately afterwards automatically.**

In the advanced settings of your chatbot, there is an option **"Disable AI"**. It is this feature that allows the user to be referred to a particular sequence when they have entered too long a text. **You can determine the number of characters beyond which this feature is triggered.**

4.2 Autonomous learning

We are talking about autonomous learning or **Machine Learning**. It is **an extremely complex process that is still limited to the world of research** and to major players in artificial intelligence such as IBM.

Botnation does not have machine learning as the researchers understand it to be, but **we do have many machine learning algorithms.**

In practice, your chatbot will determine that **certain turns of phrase probably match certain answers without you having to add a new AI rule.** For the probabilities to be more or less relevant, a large amount of data is needed.

***Example:** If you have two rules containing only the Expressions "renting brighton" and "renting london" but a user searches only for the keyword "renting" our algorithms will send it on the most popular sequence, containing at least "renting" partly in one of his Expressions.*

In any case, it is therefore advisable to **regularly check the behavior of your chatbot.**

4.3 A chatbot's AI is never over

It is important to monitor the use that your users make of it. It is only from this data that you will be able to correct / educate the ChatBot.

The vocabulary your users will use will probably be quite different from what you expected. You will therefore have to constantly teach your chatbot new expressions.

To find out which questions have remained unanswered, use the recommendation engine, which we will detail below.

***TIPS :** Botnation provides the ability to A / B Test all content on your ChatBot. We even have an automatic optimization tool.*

Online help :
[Chatbot A/B Testing Optimizations](#)

4.4 Recommendation Engine

In the AI tab of your bot, in the left column, you will find our Recommendation Engine. It consists of listing the sentences entered by users that the ChatBot did not understand and prioritizing them according to their occurrence. You can see the results for the current day, the past month, and the current and past years.

This tool therefore allows you to **identify turns of phrase and vocabulary that your chatbot has not understood** and will serve as a recommendation for adding new keywords and even new AI rules and new content.

Indeed, you will find here requests that you did not think should be processed in your ChatBot.

If you find that the same type of requests are made on a regular basis, it is time for you to address them by adding content. Whether it is responding to it in ChatBot or redirecting the user to another online service, this type of situation must be dealt with.

This is also where **you can see if your users tend to write long messages**. If this is the case, it will surely be necessary to deactivate the AI from a certain number of characters as we have seen previously.

When adding new keywords or phrases, don't forget the advice we have given you. **Analyze the sentences entered by your users and translate them for your bot.**

***TIPS :** Don't forget to update your Keyword Libraries*



NLP V2 : Next Generation

5.1 More than an evolution ...

We have developed a new version of our AI learning algorithms and mechanics.

This version is more powerful and simplifies the task of chatbot creators even more.

In the advanced settings of your chatbot is now available an option allowing you to use v1 or v2 for NLP.

All chatbots already made on Botnation are automatically configured in V1, including models. We ensure backward compatibility for all chatbots already on the platform if you don't want to change anything.

All new chatbots, and future models, will now default to V2.

Nothing prevents you from changing the configuration of an already existing chatbot v1 in v2 but its operation may require many simplifications of your rules. We recommend that you do not modify the configuration of an already operational chatbot but rather duplicate this chatbot. On this duplicated version, then change the configuration and then modify your NLP rules while thoroughly testing their new behavior before publishing this duplicated version of your chatbot.

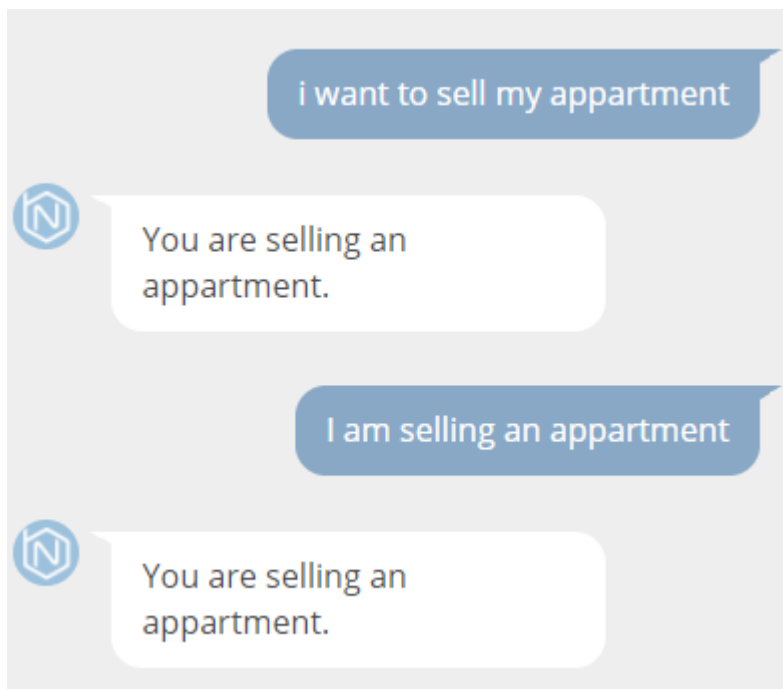
This already works in many languages and we will try to cover a dozen more quickly in the future.

5.2 Keywords and Expressions V2

With this version, you just need to **add a verb in the infinitive and all its conjugations are automatically detected**.

In addition, you only have to enter the masculine and singular words so that the plural and feminine are also detected.

Example: *We would like the AI to understand the phrases "I want to sell an apartment" and "I am selling an apartment"*



BEFORE

Using V1 you have to create a rule with the following Expressions:

sell appartement ✕ selling appartement ✕

NOW

Using V2 it suffices to use a single Expression:

sell appartement ✕

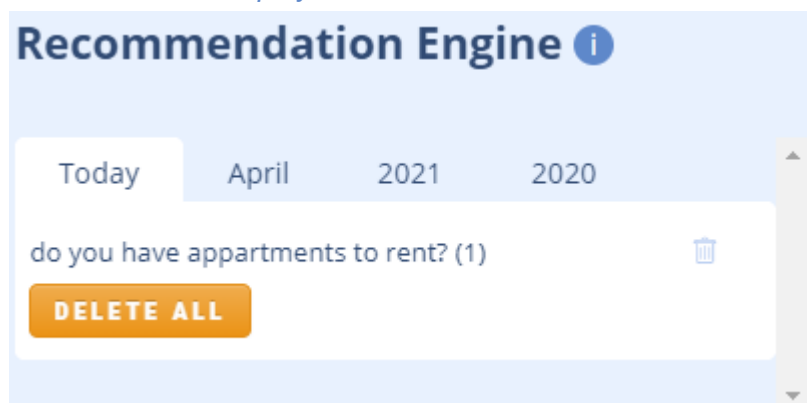
5.3 Recommendation Engine V2

In the Recommendation Algorithm tool following user input is also indicated **information between [...] the NLP's vision on this request**. This will then allow you very easily to add relevant rules and expressions.

Example: *The AI does not understand the phrase “Do you have apartments to rent?”*

BEFORE

Here is what V1 displays:



NOW

Using V2 we see that the algorithm has understood the following sequence of words and verbs: [do you apartment rent]. Which could become a rule.

You no longer need to try to think like a robot to make a new rule since you can actually see how it thinks.

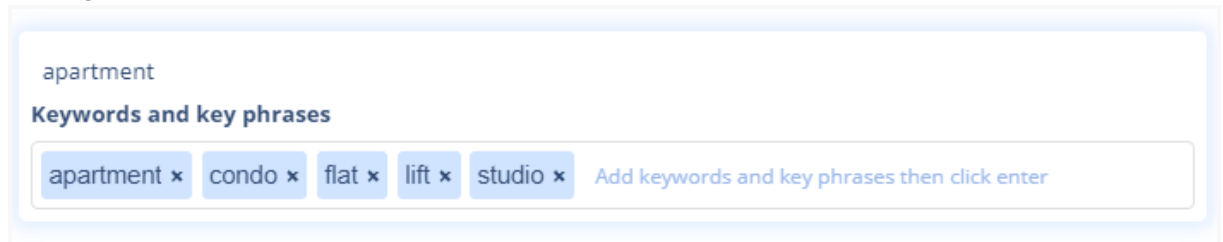


5.4 Keyword Libraries V2

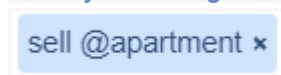
In V1, a Libraries were only usable as a kind of Keyword. This limited its uses.

Now you can include a Library in an Expression. And that changes everything.

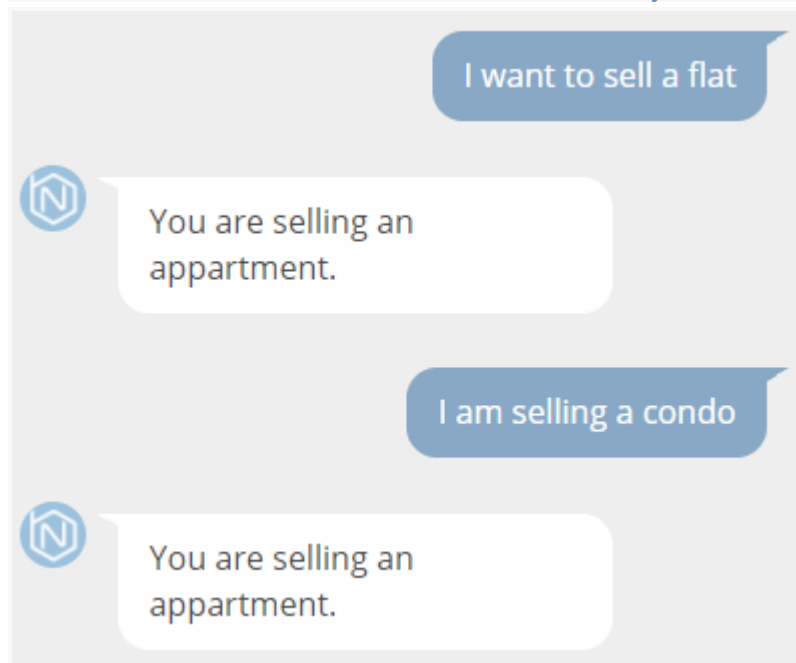
Example: By creating the following Library with synonyms for Apartment:



And by including it in this Expression in an AI Rule:



The chatbot is then able to understand and identify the following sentences:



This example perfectly shows the simplicity and the power of this Version 2 of our NLP.

Conclusion

You will understand, the AI of a ChatBot is not an element to be taken lightly and, if you decide that your project needs it, there is no miracle solution other than to devote time and brainpower.

It's exciting work and all the tools are available on our creation platform to make it happen, whether your AI is simple or very advanced.

Good luck in this adventure of Artificial Intelligence on Botnation.

So go ahead, register for free and test your AI now!

Summary of the articles cited in this document:

- [Enable / Disable user input on a sequence in my web chatbot](#)
- [How to disable the chatbot keyboard](#)
- [Transfer function - escalation - to transfer the discussion to a human](#)
- [Chatbot Templates](#)
- [Use a template to create a chatbot](#)
- [Make Your Chatbot Even Smarter with NLP](#)
- [Exit Keywords in NLP](#)
- [Introduction to the Context](#)
- [NLP's keywords libraries](#)
- [Get To Know Everything There Is About Variables {{...}}](#)
- [Chatbot A/B Testing Optimizations](#)



BOTNATION.AI

- | NO CODE | Powerful | Ergonomic | Marketing Oriented |
- | Multi-Channel | Connected to your CRM and API |
- | Used by agencies, large accounts and SMEs |
- | Rewarded with many awards | GDPR compliant |

BOTNATION THE PERFECT CHATBOT CREATION PLATFORM

botnation.ai