aito

Implement AI-driven ticket triage in the service centre without breaking the bank

#servicecenter #triage #automation

Photo by Kristine Wook

Tommi Holmgren
CPO

Service centres are ripe ground for automation, with demonstrated results to learn from and plenty of resources available to help get started. Automation benefits are obvious in customer service teams, IT supports, and anything that works based on incoming tickets or service requests.

First and foremost, the customers' happiness and loyalty increase with faster responses. Simultaneously, by removing repetitive and machine-like tasks, the work can be made more meaningful and less dull for the team's staff. Motivated team, better results across the board. Third, automation drives scalability as teams are able to serve the needs of the growing customer/user base without linear increase of the need for resources.

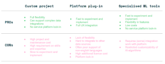
The triage's basic principle is to look at the incoming new ticket and assign it with the right **category**, **urgency** and **other information** critical for its fast and efficient processing by the right agent.

Implementation options

Assuming you have already decided to move on from full manual triage and look at the machine learning-based solutions, there are multiple ways of achieving the results, each having pros and cons. The main approaches to choose from are:

- *Custom data science project.*
  Hire a team that will build you a custom solution using state-of-the-art ML tools and the best performing algorithms just for your purpose.
- *Use plug-ins provided by the ticketing platform.*
  Most enterprise service platforms support some machine learning features today through plug-ins but are often only available at the highest licence tiers.
- *Use specialised ML-tools for the job.*
  As a middle-ground between the two above options, numerous machine learning tools can integrate with most of the service platforms and offer flexibility at a great price point.

The following table lists the main pros and cons of each of the option above.



	Custom project	Platform plug-in	Specialised ML tools
PROs	• Full flexibility • Can support complex data integrations • No service platform lock-in	• Fast to experiment and implement • Full UX integration	• Fast to experiment and implement • Flexibility in features • Low costs • No service platform lock-in
CONs	• High project and maintenance cost • High requirement on skills and expertise • Slow to experiment and implement	• Lack of flexibility • Hard to integrate to other data sources • Often poor support of non-english languages • High additional licence cost • Platform lock-in	• Requires (some) integration work with platform • Restricted customizability of AI-algorithms

Comparison of pros and cons of different implementation options

In my experience, there is a fundamental question of what is prioritised: flexibility vs costs vs speed.

If you want more than just a service centre solution and look for integrating loads more than ticketing triage (and the cost is not an issue), maybe you are in a position to hire a data science team to build the tools tailored for you. The benefit here is also that you are independent of the service platform with your AI-features and can bring them to other platforms if you later choose.

If you want to spend as little time as possible and are happy with the restrictions of the tools offered by your platform provider (e.g. in languages and features), go with the plug-ins provided by the platform. But be ready for a price-hike. For example, with ServiceNow, the AI features are available at the higher pricing tiers, known to have additional cost in the range of +50% per agent. In Freshdesk, you'll need to be on the highest pricing tier, which is +100% price increase from the previous tier.

If you have some development resources at hand and want a solution with speed and flexibility yet where you have better control of the costs, look at the ML-tools available in the market. You will also retain your AI work's portability, not being bound to the service platform in use today. The rest of the post takes a deeper look at implementing ticket triage at an IT service desk using Aito.ai-a simple yet powerful ML tool for automation engineers.

This case has been developed with the friends at Futurice using their IT service ticket data from FreshDesk as an example (personal data is naturally hidden). The case only looks at the machine learning features to keep it at a reasonable length. There are plenty of resources available for integrating with the Freshdesk, and even some of the automation platforms like Integromat can do the job.

Ticket triage at Futurice with Aito.ai

There is no one and the only right way to do the triage. Instead, it would be best if you started defining your objectives and only then looking at the steps needed to achieve them. The examples below represent steps taken at Futurice IT support, where the amount of tickets has grown along with the blooming business so much that it was time to take the first steps with machine learning assisted ticket triage.

> ***Helping people to succeed via harnessing organisational knowledge is our key value lever.*** *-- Tuomas Syrjänen, Data & AI, Co-founder, Futurice*

Preparations take place first. Historic tickets were exported from FreshDesk and uploaded to Aito.ai in the UI. Also, the new tickets are periodically exported and added into Aito's dataset to keep the training data fresh. Aito does not require any additional steps like model re-training or deployment but is always "fresh" based on the current data.

First, tickets are categorised and subcategorised according to the structure already defined in FreshDesk. Previously this job was manual, meaning that each ticket had an agent touch for triage. Categorisation helps in the tickets' assignment to the right place, monitoring, and planning processes' improvements. With each incoming ticket, there is a query to Aito to predict the category with confidence values.

```
query = {
    "from": TABLENAME,
    "where": {
        "subject": "Laptop issues",
        "requesterId": "XXXXXXXXXXX",
        "sourceName": "Portal",
        "description": "My computer is restarting at random intervals, even after upda
    },
    "predict": "category",
    "limit": 2
}
```

The above query gets us two top predictions for the category.

```
{
  "hits": [
    {
      "$p": 0.739823420978263,
      "feature": "Hardware",
      "field": "category"
    },
    {
      "$p": 0.06230314322171832,
      "feature": "Mobile Budget",
      "field": "category"
    }
  ],
  "offset": 0,
  "total": 19
}
```

When knowing that we are likely (over 70% confidence) dealing with a Hardware-related request, we can still make another query and see if we can assign this to a subcategory. Query already looks familiar but uses additional from-where block, in the beginning, to restrict the training data to only within this category's tickets.

```
query = {
    "from": {
        "from": TABLENAME,
        "where": {
            "category": "Hardware"
        }
    },
    "where": {
        "subject": "Laptop issues",
        "category": "Hardware",
        "requesterId": "XXXXXXXXXXX",
        "sourceName": "Portal",
        "description": "My computer is restarting at random intervals, even after upda
    },
    "predict": "sub-category",
    "limit": 2
}
```

Again, the result gives us the most likely sub-categories. It is Computer-related and can confidently write these categories back to the ticket data.

```
{
  "hits": [
    {
      "$p": 0.962468610758821,
      "feature": "Computer",
      "field": "sub-category"
    },
    {
      "$p": 0.024528328466996626,
      "feature": "non-categorised",
      "field": "sub-category"
    }
  ],
  "offset": 0,
  "total": 5
}
```

Second, we predict the urgency of the ticket based on the three-tier urgency (1, 2, 3) at use in the team. You are getting hold of this; look how similar the query look, just with a few twists! :)

```
query = {
    "from": TABLENAME,
    "where": {
        "subject": "Laptop issues",
        "requesterId": "XXXXXXXXXXX",
        "sourceName": "Portal",
        "category": "Hardware",
        "sub-category": "Computer",
        "description": "My computer is restarting at random intervals, even after upda
    },
    "predict": "urgency",
    "limit": 2
}
```

The response gives the most likely urgency, and this is again easy to write back to the FreshDesk platform to help the agents prioritise their work.

```
{
  "hits": [
    {
      "$p": 0.8998651665320588,
      "feature": "2",
      "field": "urgency"
    },
    {
      "$p": 0.09132182836330256,
      "feature": "1",
      "field": "urgency"
    }
  ],
  "offset": 0,
  "total": 3
}
```

Third, we will look for similar cases in the past and what has been the response to them. There are numerous ways to develop this further! For example, if your team uses macros in replying to requests, instead of finding a match of a previous reply, you could instead look for the best macro to be used for the request and help the agent automate even more of the work.

Macros were not yet in use at the team. Thus we look at the top 3 precious responses to similar requests and link them for agents benefit. The query goes like this:

```
query = {
    "from": TABLENAME,
    "where": {
        "subject": "Laptop issues",
        "requesterId": "XXXXXXXXXXX",
        "sourceName": "Portal",
        "category": "Hardware",
        "sub-category": "Computer",
        "description": "My computer is restarting at random intervals, even after upda
    },
    "match": "response",
    "limit": 3
}
```

The response gives the best possible matches (output is concatenated only to contain one to keep it readable).

```
{
  "hits": [
    {
      "$p": 0.4999865856715547,
      "$value": "Hi XXXXX, I had the same issue couple of days back. I suggest you to
      Please notice that some issues have been discovered. Upgrade only if you do not
      Notice that in the latest version of Big Sur 11.2 some issues have been discover
      No functioning and reliable fix or workaround has been discovered for either pro
    },
...
  ],
  "offset": 0,
  "total": 3408
}
```

If I were working on the case, that would be pretty helpful and definitely shorten the time it takes to reply. The right link is already there, no need to look for it separately.

As the last step, we can try assigning the ticket. There is a disclaimer, though. This step is way more complex than it first sounds. Finding the "right" agent for the ticket brings questions about workforce and shift planning (is she free now?) and about keeping the work motivating. Maybe you don't want to answer the same tickets over and over again, even if you would be the best at it. Some companies would assign the ticket to the right team or queue. Often that can already be done in the platform using the categories we already predicted and would not need additional Aito query.

But we wanted to add a bit more flavour here. We wanted to attach information to the ticket about which agent has replied to similar tickets the most. Whoever gets assigned the ticket has this information at hand, should they need assistance. This is again a query away!

```
query = {
    "from": TABLENAME,
    "where": {
        "subject": "Laptop issues",
        "requesterId": "XXXXXXXXXXX",
        "sourceName": "Portal",
        "category": "Hardware",
        "sub-category": "Computer",
        "description": "My computer is restarting at random intervals, even after upda
    },
    "predict": "responder-name",
    "limit": 3
}
```

The response looks like this, with personal details of the agents hidden. Yet again, this is easy to add to the ticket data.

```
{
  "hits": [
    {
      "$p": 0.590856133747189,
      "feature": "AGENT-Y",
      "field": "responder-name"
    },
    {
      "$p": 0.19139728528795614,
      "feature": "AGENT-Z",
      "field": "responder-name"
    }
  ],
  "offset": 0,
  "total": 25
}
```

This concludes the steps of the triage (and enrichment). Using the information provided by Aito.ai queries, the team can continuously automate and optimise the operations and are not limited to only the features described here.

I hope you have enjoyed the example, and it helps you get ahead with your service centre automation journey! You can reach me best at the Aito.ai Slack group or via Twitter.