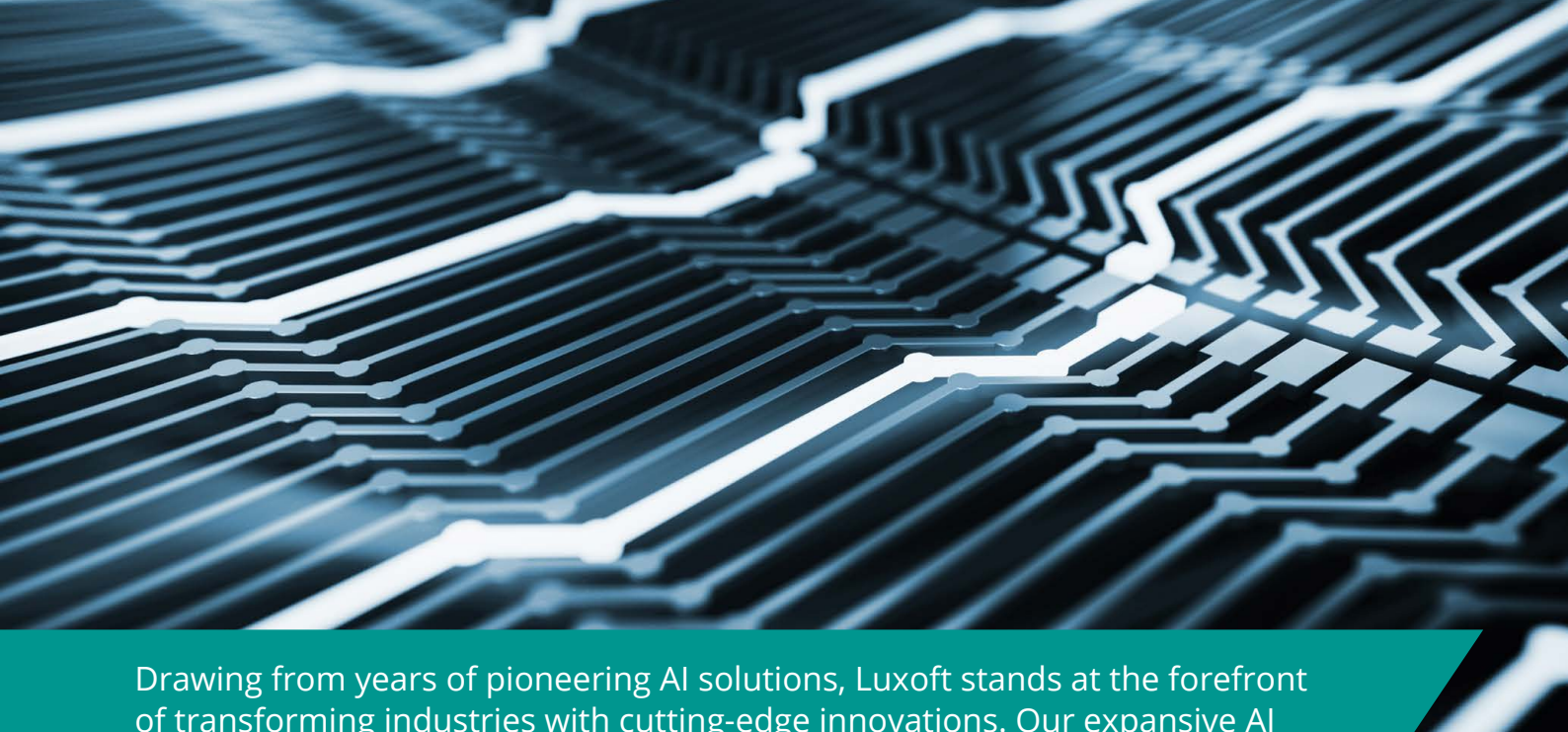


#AIJournal

Automated testing with AI — part of the triage process

by Artur Shaikhatarov, Project Manager, Automotive, Luxoft
and Martin d'Ippolito, Software Developer, Automotive, Luxoft



Drawing from years of pioneering AI solutions, Luxoft stands at the forefront of transforming industries with cutting-edge innovations. Our expansive AI expertise has consistently delivered unparalleled results for clients worldwide.

In brief

- During the triage process, we need to determine whether two documents, sentences or chunks of text, have the same meaning. In this case, we needed to determine whether new Jira tickets were duplicating tickets that were already on the platform
- The power of machine learning doesn't just lie in the models. It's about understanding data characteristics during exploratory analysis and arranging the data so the model performs consistently well in the real world
- Luxoft's GEMS team trains models on data they have to hand, which is passed on to the model. Because they don't know what it will be like in advance, they use ML and data science to resolve the issue

In this technical article on projects executed by Luxoft's Graphics Embedded Machine Learning Simulation (GEMS) team, we'll review a video testing automation initiative commissioned by a major motor manufacturer.

This second project deals with duplicate search and had a greater impact on the business than the subject of our previous article. The issue resolved was around the detection of duplicate sentences. In other words, understanding whether two documents, sentences or chunks of text had the same meaning.

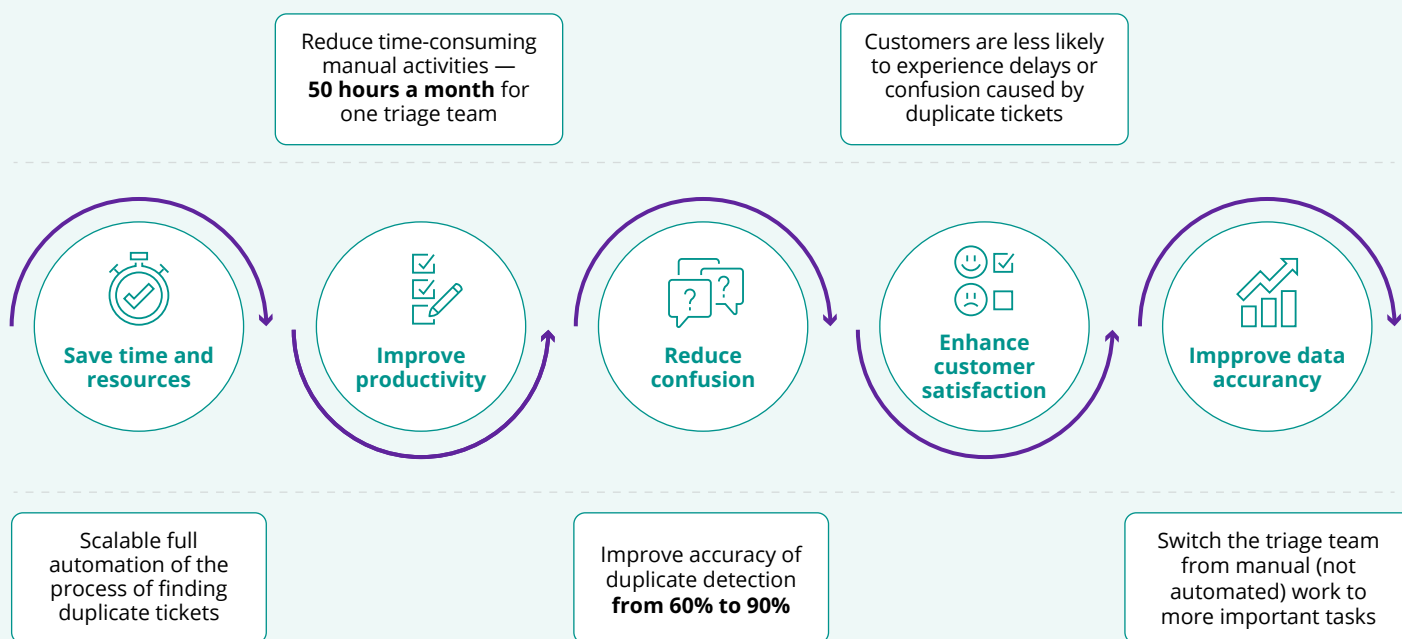
In this particular case, we needed to determine if newly raised Jira tickets could be duplicating existing tickets (Jira is a management platform used by teams to plan, track and manage tasks and issues.) Through completing this task successfully, the GEMS team demonstrated how automation could reclaim the time wasted not just on duplicated tickets but on manual processes in general.

Making up for lost time

It might not appear on the balance sheet, but time is a core business asset, nonetheless.

So, conserving this precious commodity has to be a major priority for all leaders and decision-makers. You won't find the advantages of implementing AI-powered automated duplicate search listed in "[A la recherche du temps perdu](#)," perhaps. Still, they focus on the benefits of making up for lost time all the same.

For example, let's say we have 1,000 tickets; each ticket requires 40 minutes, and there are 60 duplicates. That's a massive saving of 5 working days. By automating processes, this machine learning application has a significant, positive impact on human and financial resources.



The application improves productivity, reduces multi-ticket confusion and enables the business to be more agile. Consequently, customer satisfaction increases, and the Jira platform data is streamlined and more

reliable. By the way, this is not for Jira platforms exclusively. We can deploy a sentence duplicate mechanism on any system with tickets and actual information.

Different people see different similarities and similar differences

Vladimir Nabokov

"Not all those who wander are lost."

George Orwell

"Abandon all hope, ye who enter here."

Dante Alighieri



*"All that we see or seem is
but a dream within a dream."*

Edgar Allan Poe

Sentence
similarity

*"We are such stuff as dreams are made on,
and our little life is rounded with a sleep."*

William Shakespeare



*"Freedom is the
right to tell people
what they do not
want to hear"*

George Orwell

Sentence
similarity

*"I may disagree with what
you have to say, but I shall
defend to the death your
right to say it."*

Voltaire



LEGEND:

Text #1

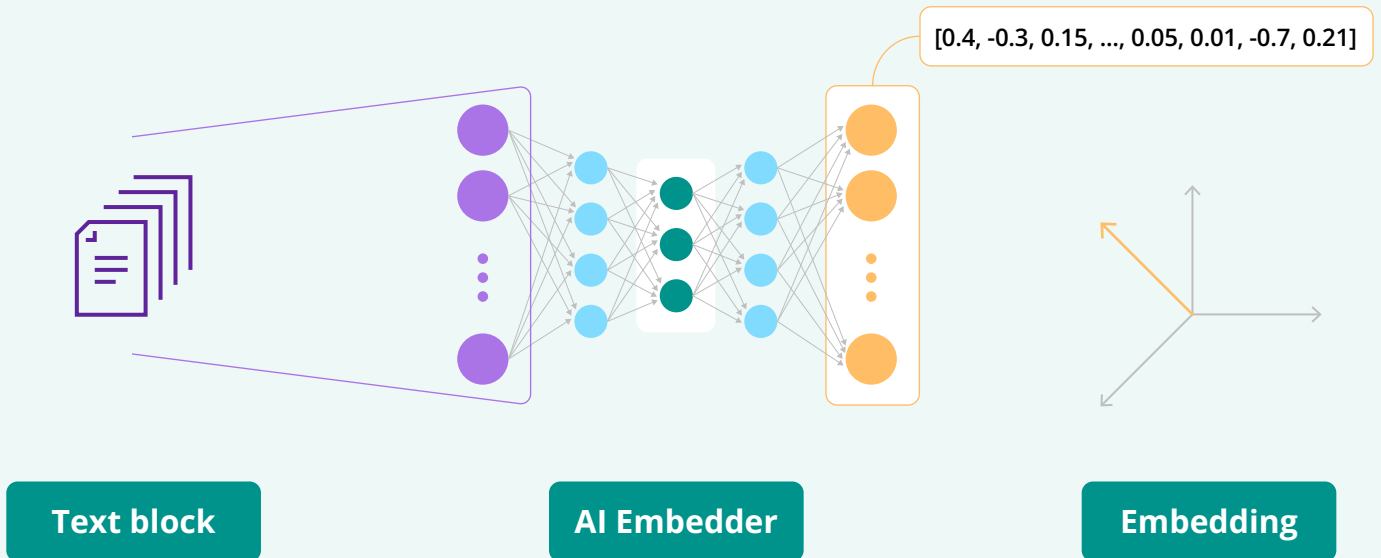
Text #2

Degree of similarity
between text #1 and text #2

Before we go too far, we need to establish what we mean by "meaning." The similarity of the first two texts (Orwell and Dante, above) is zero; there's no overlap. Some texts have similar meanings, but the similarity is low. Other texts have more or less the same meaning but are written differently. Now, we're dealing with

two types of similarity: Lexical similarity (words are important) and semantic similarity (meaning is more important).

So, we implement embedding — a leading-edge machine learning model for dealing with text.

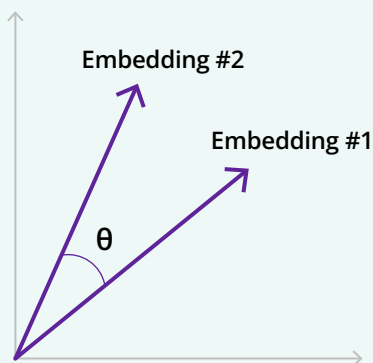


Text is passed through a machine learning neural network called Embedder and translated into a computer machine. The machine doesn't understand text, only numbers, which is fine because embedding works with vectors — collections of numbers or coordinates (so the correct number order is essential).

The Luxoft GEMS team has the right skills to work with any of the NLP models, but for this application,

we needed the Transformer S-BERT (Hugging Face). We experimented with ULMFiT and SimCSE, too, but S-BERT had the most suitable mix of capabilities.

Once we had two embeddings (two lists of numbers we added), we used a cosine similarity metric to work out how to compute the vector similarity.

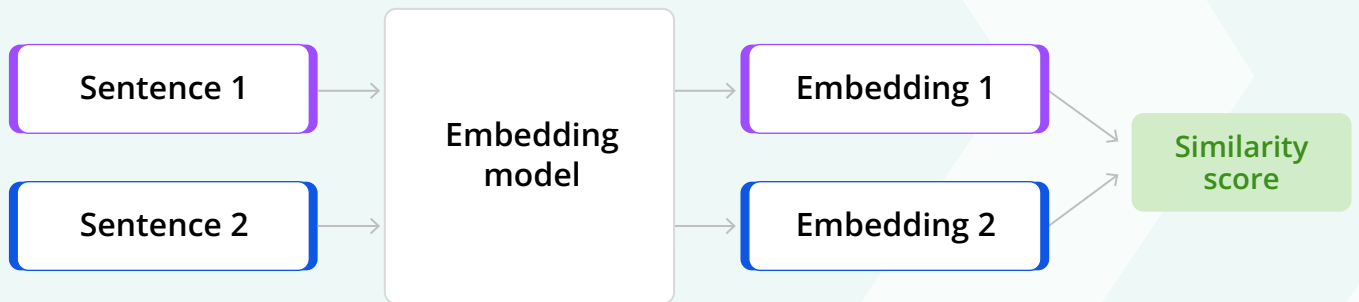


$$\text{Embedding}_1 = \begin{bmatrix} x_1, \\ x_2, \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix}$$

$$\cos(\theta) = \text{similarity}(\text{Embedding}_1, \text{Embedding}_2) =$$

$$\frac{\text{Embedding}_1^T \text{Embedding}_2}{|\text{Embedding}_1| |\text{Embedding}_2|}$$

And this is how we did it in practice:



Two sentences passed through the embedding model. We had two in embedding and computed the similarity.

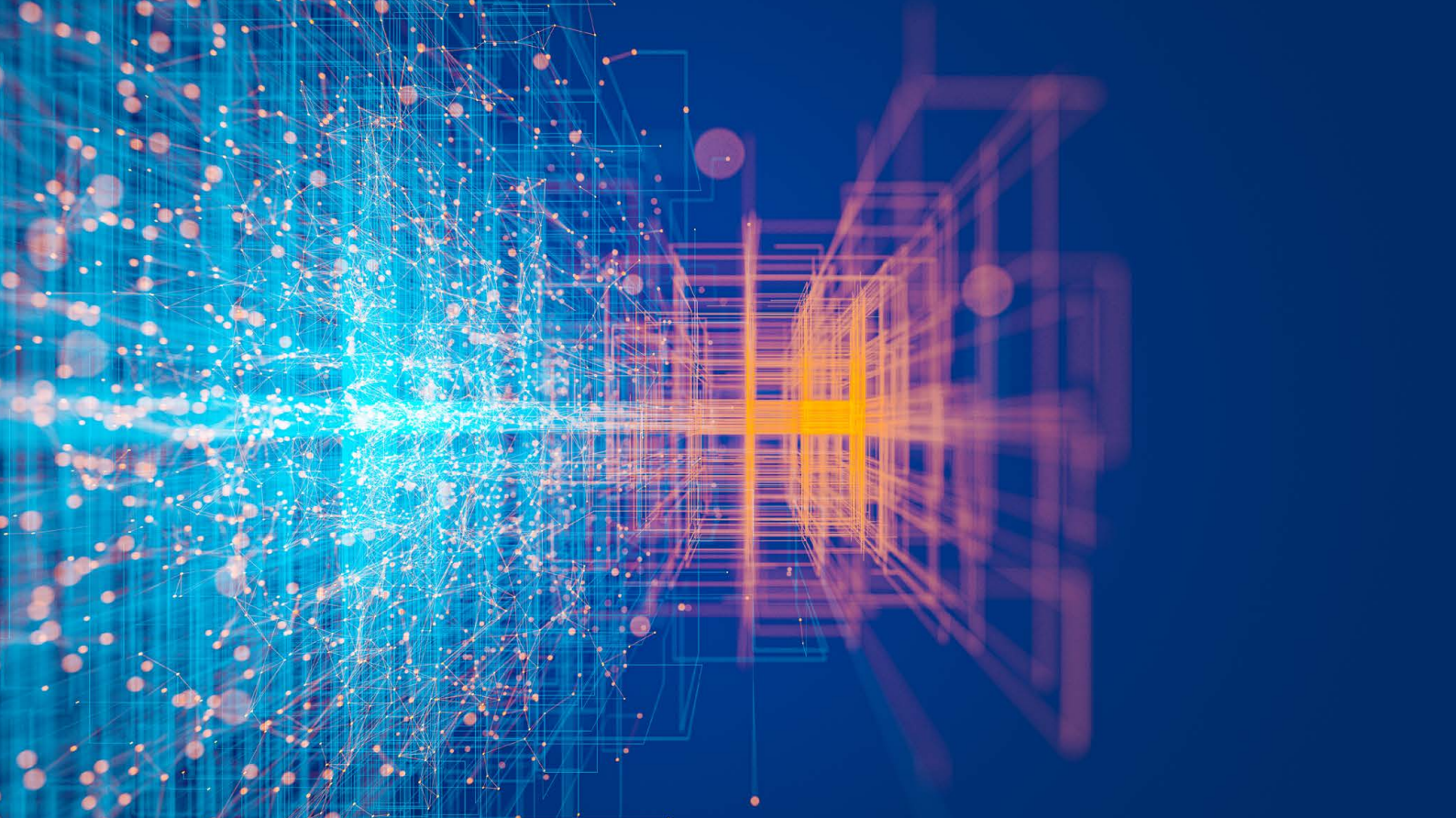
Client challenge

To recap, the client approached Luxoft with a duplicates problem. We completed the data processing, downloaded the tickets and built a correct data set, which is the most critical task.

First, we used Sentence-BERT, our best model. Second, we built a data set that reflected the data characteristics. Why? Because another machine learning practitioner might be tempted to run a model with any old data and expect the model to work perfectly. But

the power of machine learning doesn't lie in models exclusively. It lies in understanding data characteristics during exploratory analysis. It's also about arranging the data so the data-trained model will perform consistently well in real-world business applications.

We collected a sample of duplicate tickets and gave them a similarity score of one. Non-duplicate tickets got a similarity score of zero (a good compromise for minus one — exactly opposite to one).



Preparation is essential

Much data preprocessing was undertaken to prepare our data set for the training model. Once completed, we split the data set (training and testing). The training information was passed to the model to determine the data characteristics. The testing data set told us whether the trained model would work in a real-world scenario.

Why was this necessary? Because we see good performances on the training data set, but that's expected because the model was trained on this data. However, the model has never seen the testing data set tickets. We train models on data we have to hand, and the new data is passed to the model. Of course, the team doesn't know in advance what it will be, so we use machine learning and data science.

Once again, we took the issues, cleaned them, arranged them into duplicates and non-duplicates and trained the model with the data.

We could have done things locally, but for intensive training, we would have needed a GPU machine, in this case, an NVIDIA Tesla K80 dual-GPU accelerator card. Often, it's impossible to fine-tune very large models (or specialize this model with our data.) The most prominent model (all-mpnet-base-v2) was too big because it was pre-trained on a massive amount of data. And since the performance was unlikely to improve, training would have proved uneconomical.

Once we've trained a model, we have four cases: True positives (the model understands actual duplicate tickets) and true negatives (the opposite). We also have false positives (similar, but not duplicates) and false negatives (duplicates, but the model doesn't know that).

Our role was to minimize the false positives and negatives and achieve the highest metrics possible. Machine learning models can never be 100% accurate, so we have to have a trade-off between some of the metrics. The most important metric is the recall, where the model looks to detect most of the duplicates.

Cross-industry solution

The most important thing here is to have the Jira administrator annotate which tickets are duplicates.



A good recall confirms that the trained model is a good instrument for working with duplicates

Drawing on our domain expertise, we can query a client's Jira database, construct a data set and use the data set to train a model. Then, we can integrate this crucial duplicate data into any system, whether banking, the automotive sector or whatever. From a

practical business perspective, once we've marked the duplicate tickets and improved the triaging processes, staff can ignore them and focus on productive work, reclaiming lost time, reducing stress levels and, more importantly, cutting costs.

But how much does the client save in hard cash?

The total depends on the number and characteristics of their tickets. Nevertheless, here's a rough estimation of the ROI:

- The cost of supporting a classic duplicate-ticket-checking team is about €251,000 a year
- Developmental costs are around €80,000 (one-time investment)
- The client saves about €225,000 a year $(225,000 - 80,000) / 251,000$ which equals 58% ROI

The real savings are much higher as the €80,000 developmental cost is a one-time investment, and €225,000 is an annual saving for life. Also many of the automation-of-manual-testing models can be used again, both for this client and elsewhere.

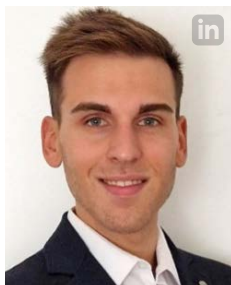
About **the authors**



Artur Shaikhatarov

AI/ML Project Manager, Automotive, Luxoft

Arthur has served as project manager specializing in AI/ML initiatives at Luxoft for 4 years. His journey in artificial intelligence and machine learning began as an engineer, where he honed his skills and knowledge. Arthur is responsible for team management and formation, presales activities for ML projects, the orchestration and execution of AI/ML proofs-of-concept and the growth of AI/ML activities within Luxoft.



Martin d'Ippolito

Data Scientist and ML Engineer, Automotive, Luxoft

Martin is an accomplished data scientist and machine learning engineer, with five years' expertise in the field. His knowledge and experience go beyond computer vision and natural language processing (NLP), to include the development of end-to-end web applications. This unique combination of talents helps his team craft outstanding proofs-of-concept to engage clients.

Let's touch base

If you'd like to learn more about the practicalities of automating duplicate search, visit luxoft.com or [contact us](#).

About Luxoft

Luxoft, a DXC Technology Company delivers digital advantage for software-defined organizations, leveraging domain knowledge and software engineering capabilities. We use our industry-specific expertise and extensive partnership network to engineer innovative products and services that generate value and shape the future of industries.

For more information, please visit luxoft.com