# Approaching the Text2SQL challenge

## A conversational experience for digital twins

Janna Lipenkova | ARIC Brownbag, November 1, 2022

# AGENDA

1. **Background: building a Digital Twin for software companies**
2. The Text2SQL task and challenges
3. Our workflow for optimizing Text2SQL
4. Using Large Language Models with constrained decoding
5. Discussion & how to get involved

# Problem: software companies use over 100 apps today …

## Marketing

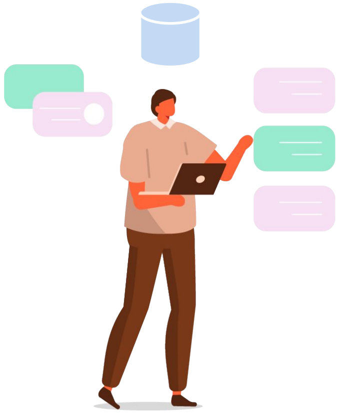
## Sales

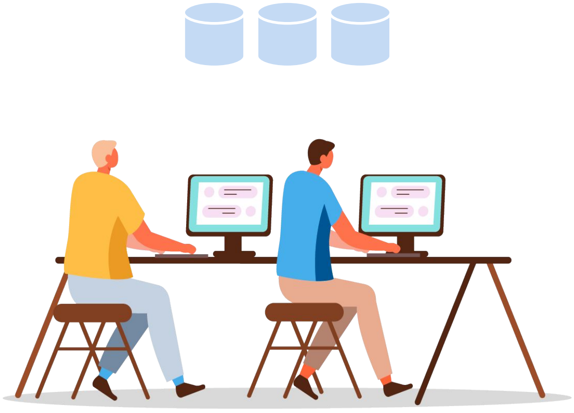
## Success

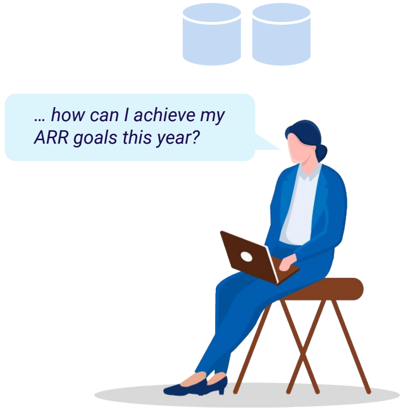
LinkedIn Marketo An Adobe Company Google Analytics HubSpot Calendly GONG salesforce Outreach in SALES NAVIGATOR Gainsight INVIDED Appcues skilljar

sproutsocial SEMRUSH 6sense zoominfo Segment Endgame Clearbit Clari Jira kibana zoom Zendesk slack qualtrics XM

CRAYON craft.io mixpanel Figma Prometheus Jenkins GitHub DATADOG Grafana Outlook tableau NETSUITE Microsoft Teams







## Product

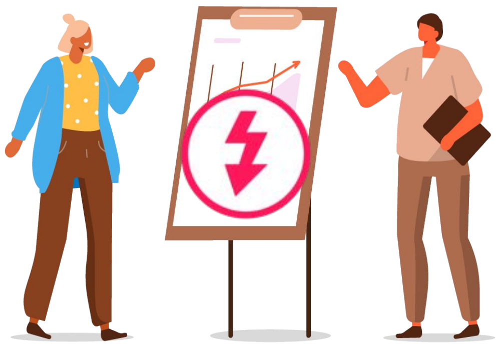
## Engineering

## Management

# … too much data for effective human decision-making

Marketing

Sales



Success

ACTIVITY DATA
- Web clicks
- Product clicks
- Customer feedback
- Competitor activities
- Emails & slack
- JIRA tickets
- Developer PRs
- …

HUMAN LIMITATION

What does all the data tell us about the best actions to achieve our goals?

Problem: firms generate Mbits of data, but humans only have a cognitive bandwidth of 40 bit/s

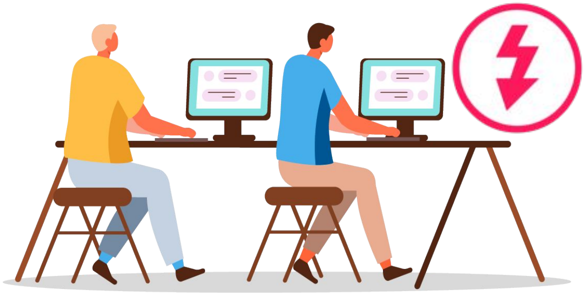
HUMAN BEHAVIOUR

In 90% of decisions, humans cherry-pick data that supports their intuition

IMPACT

Decision making is hard & error prone

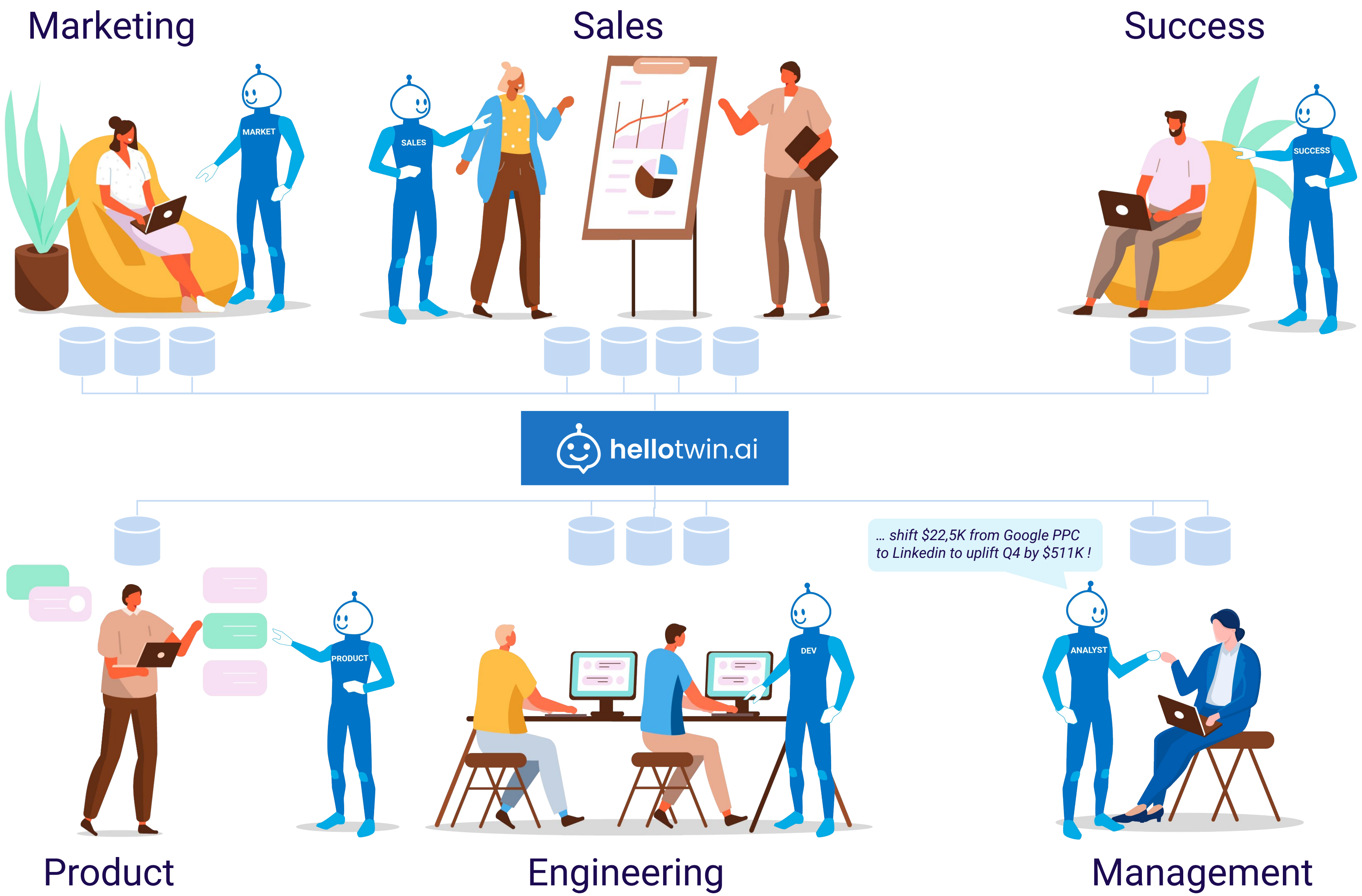
Product



Engineering

Management

# Solution: hire Digital Workers (“Twins”) as teammates



Marketing

Sales

Success

hellotwin.ai

… shift $22,5K from Google PPC to Linkedin to uplift Q4 by $511K !

Product

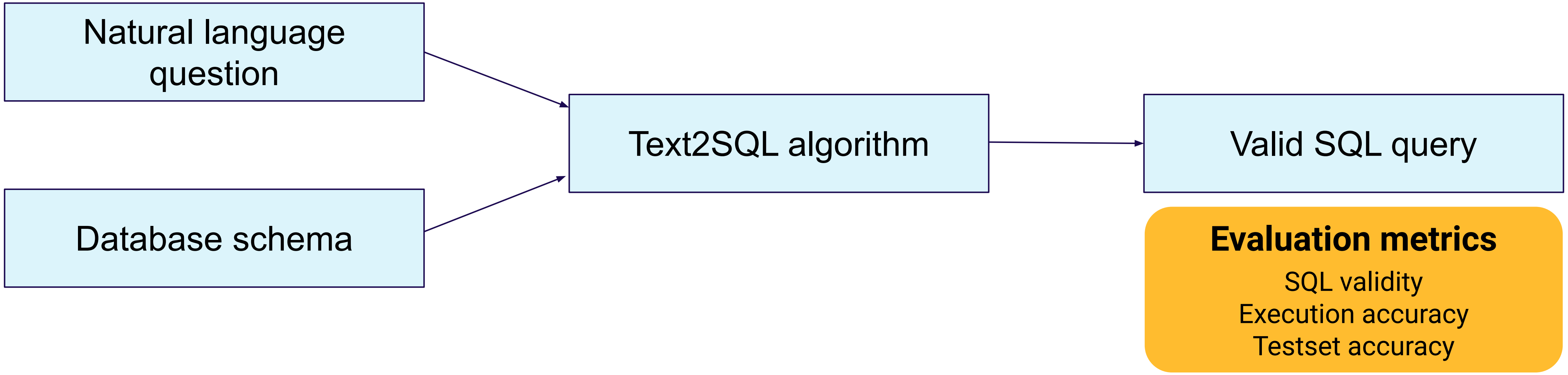
Engineering

Management

# AGENDA

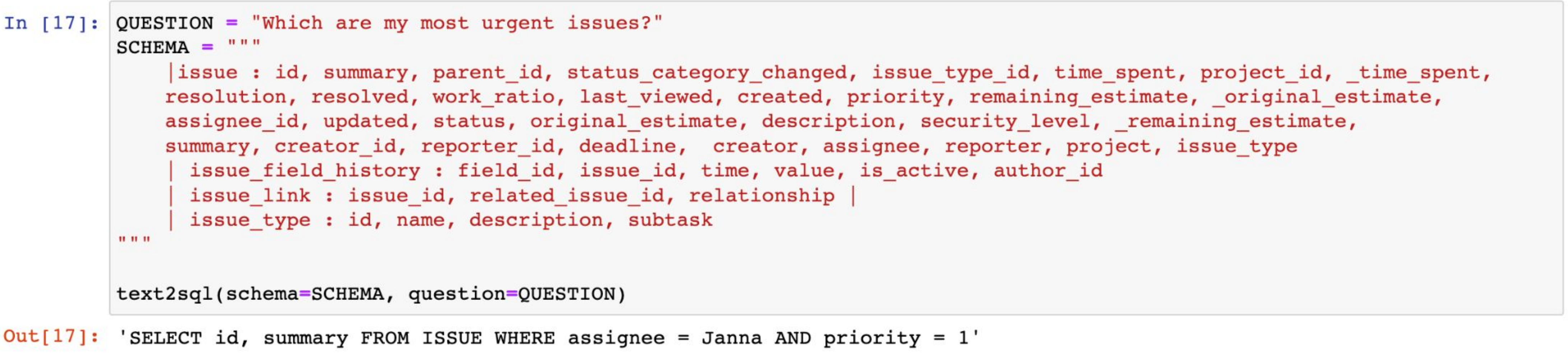
1. Background: building a Digital Twin for software companies
2. **The Text2SQL task and challenges**
3. Our workflow for optimizing Text2SQL
4. Using Large Language Models with constrained decoding
5. Discussion & how to get involved

# The Text2SQL challenge



**Evaluation metrics**

SQL validity
Execution accuracy
Testset accuracy



```
In [17]: QUESTION = "Which are my most urgent issues?"
         SCHEMA = """
             |issue : id, summary, parent_id, status_category_changed, issue_type_id, time_spent, project_id, _time_spent,
             resolution, resolved, work_ratio, last_viewed, created, priority, remaining_estimate, _original_estimate,
             assignee_id, updated, status, original_estimate, description, security_level, _remaining_estimate,
             summary, creator_id, reporter_id, deadline,  creator, assignee, reporter, project, issue_type
             | issue_field_history : field_id, issue_id, time, value, is_active, author_id
             | issue_link : issue_id, related_issue_id, relationship |
             | issue_type : id, name, description, subtask
         """

         text2sql(schema=SCHEMA, question=QUESTION)

Out[17]: 'SELECT id, summary FROM ISSUE WHERE assignee = Janna AND priority = 1'
```

# Text2SQL is not yet solved at the industry level

**Natural Language Understanding challenges**

- SELECT fields
- JOINs
- Aggregations and WHERE-filters

**SQL generation challenges**

- Ensure SQL validity
- Generalization to new database schemas / schema changes
- Calibrate use of world knowledge by Large Language Models (e.g. *issue* -> *task* or *bug*?)

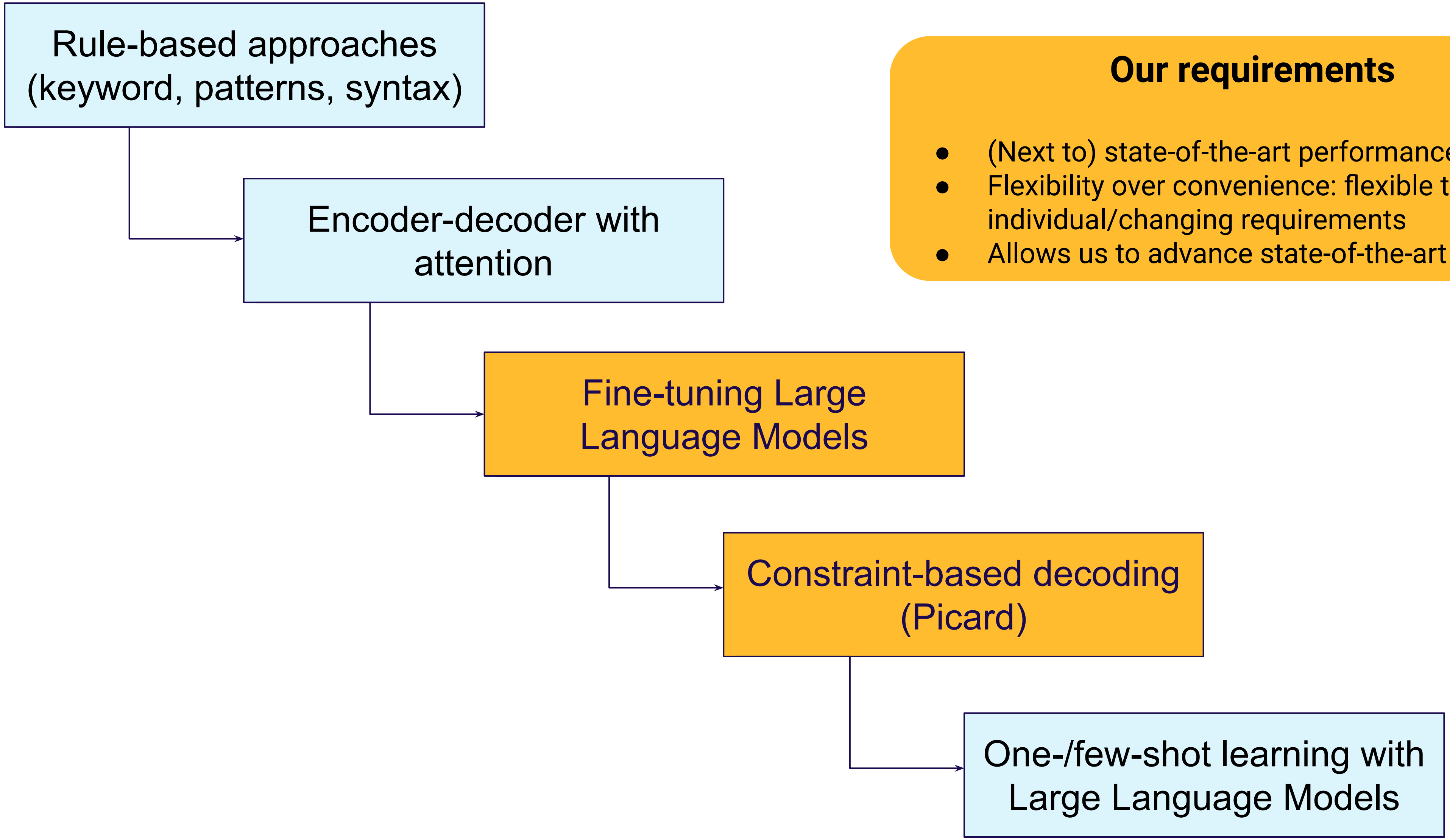
**Conversational challenges**

- Inaccurate or ambiguous inputs
- Context dependency and coreference resolution

**SOA Text2SQL approaches exhibit 76-79% testing accuracy on the Spider benchmark*.**

*Yu et al. (2018): Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task

# Major Text2SQL approaches



Rule-based approaches (keyword, patterns, syntax)

Encoder-decoder with attention

Fine-tuning Large Language Models

Constraint-based decoding (Picard)

One-/few-shot learning with Large Language Models

**Our requirements**

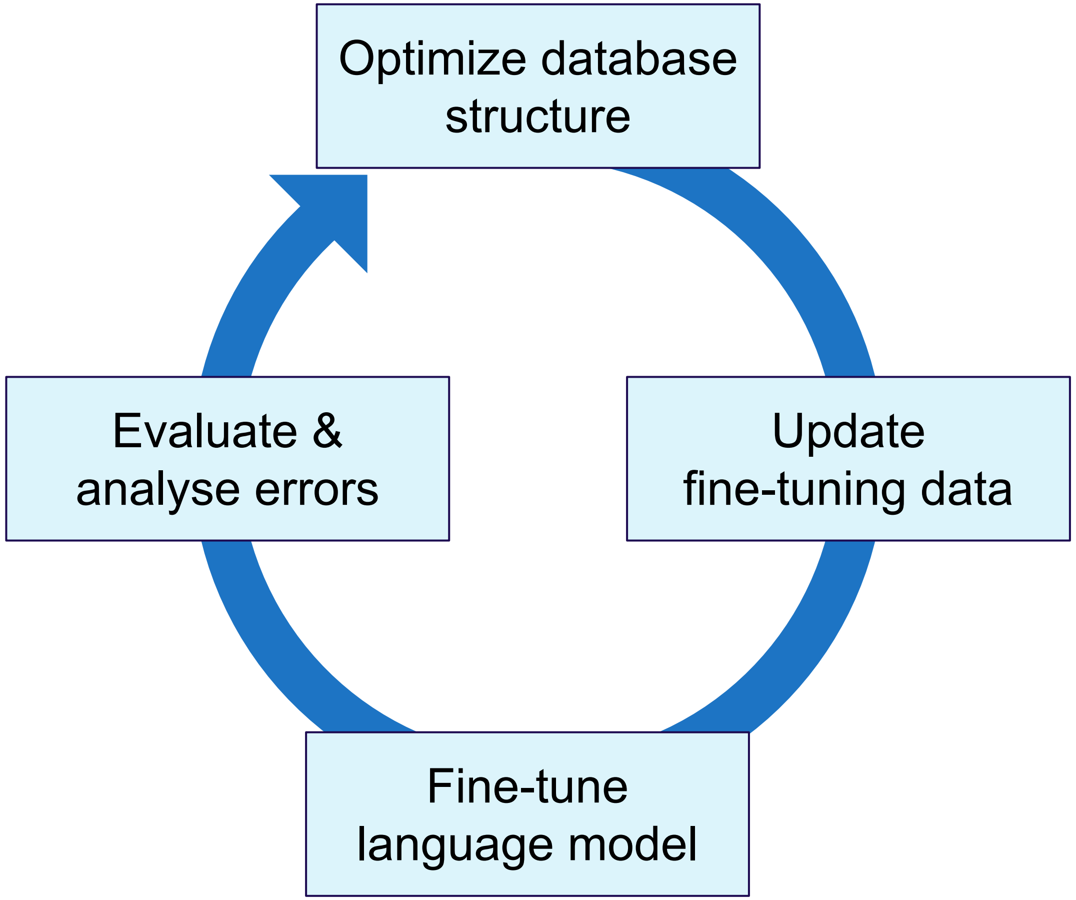
- (Next to) state-of-the-art performance
- Flexibility over convenience: flexible tuning to individual/changing requirements
- Allows us to advance state-of-the-art over time

# AGENDA

1. Background: building a Digital Twin for software companies
2. The Text2SQL task and challenges
3. **Our workflow for optimizing Text2SQL**
4. Using Large Language Models with constrained decoding
5. Discussion & how to get involved

# Our workflow for optimizing Text2SQL

- Natural, readable column and table names
- Use of “wide” tables to reduce the need for JOINs
- Pre-computation of common aggregations



Optimize database structure

Update fine-tuning data

Fine-tune language model

Evaluate & analyse errors

- Main metric: execution accuracy
- Error analysis by SQL challenge tags

- Add and expand frequent error cases
- Adjust labels (SQL queries) to new DB schema

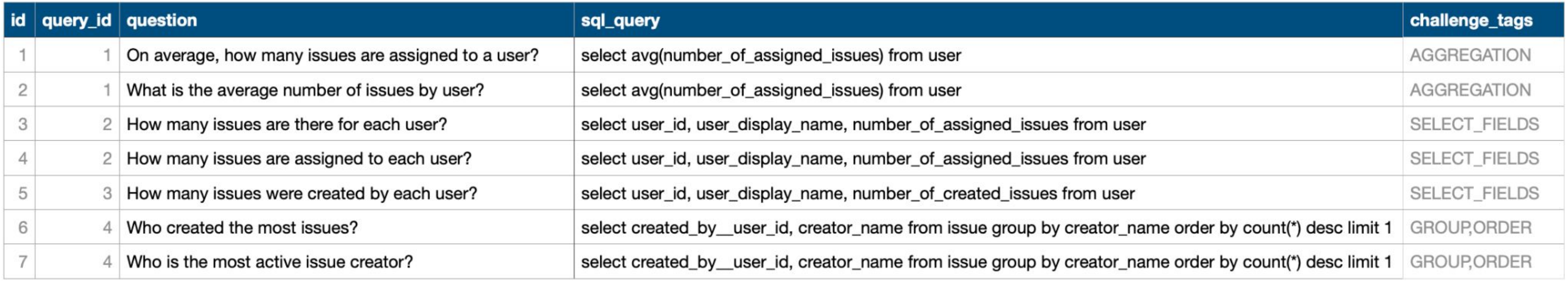
Using full fine-tuning data or focus on specific challenges

# AGENDA

1. Background: building a Digital Twin for software companies
2. The Text2SQL task and challenges
3. Our workflow for optimizing Text2SQL
4. **Using Large Language Models with constrained decoding**
5. Discussion & how to get involved

# Building Text2SQL fine-tuning data

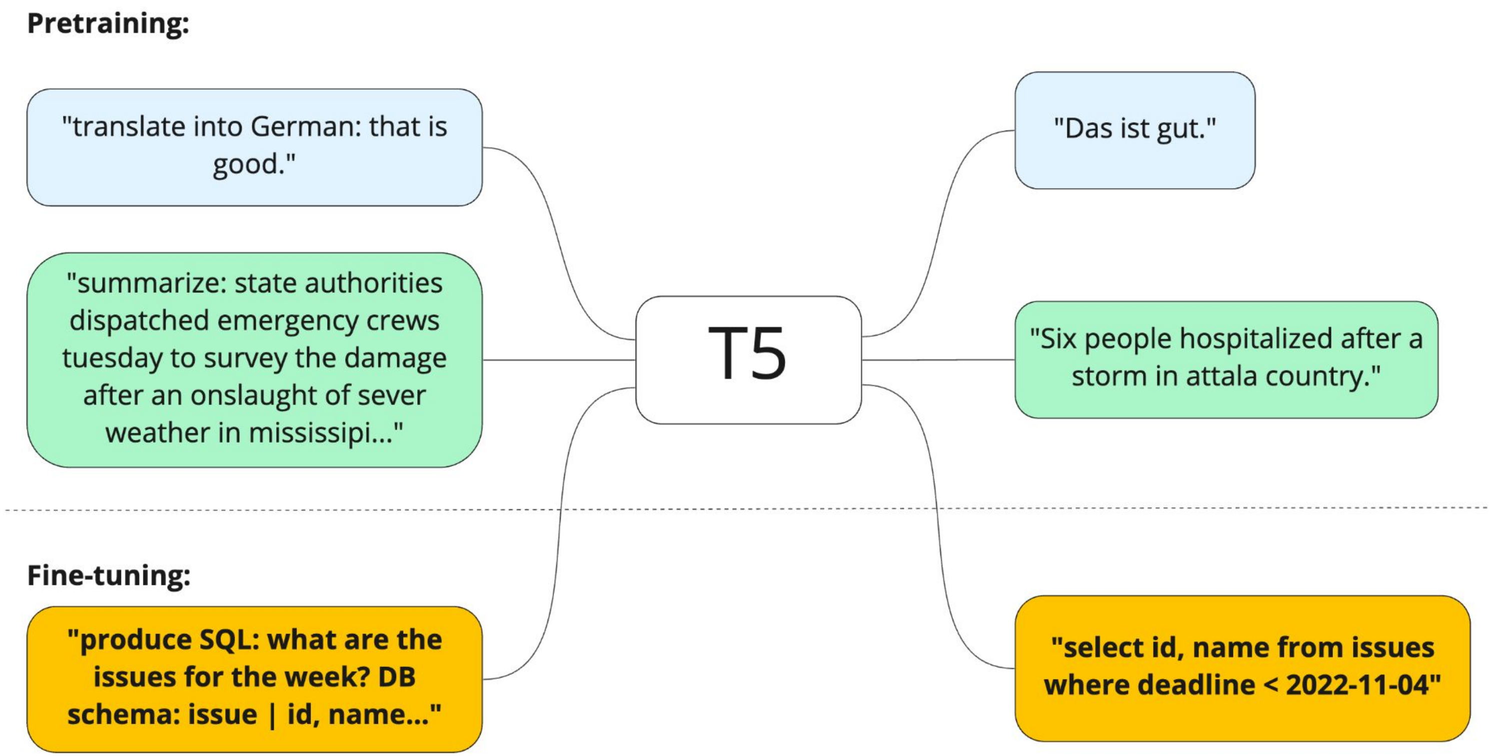
- We construct a **specialised fine-tuning dataset** of (question, SQL query) pairs
- **One-to-many mapping** from SQL queries to questions; paraphrases are generated using GPT-3
- Additional tagging by **major challenges** (AGGREGATION, JOIN etc.) for more targeted fine-tuning
- Regular updates using templating to adjust to changes in database structure



| id | query_id | question | sql_query | challenge_tags |
|---|---|---|---|---|
| 1 | 1 | On average, how many issues are assigned to a user? | select avg(number_of_assigned_issues) from user | AGGREGATION |
| 2 | 1 | What is the average number of issues by user? | select avg(number_of_assigned_issues) from user | AGGREGATION |
| 3 | 2 | How many issues are there for each user? | select user_id, user_display_name, number_of_assigned_issues from user | SELECT_FIELDS |
| 4 | 2 | How many issues are assigned to each user? | select user_id, user_display_name, number_of_assigned_issues from user | SELECT_FIELDS |
| 5 | 3 | How many issues were created by each user? | select user_id, user_display_name, number_of_created_issues from user | SELECT_FIELDS |
| 6 | 4 | Who created the most issues? | select created_by__user_id, creator_name from issue group by creator_name order by count(*) desc limit 1 | GROUP,ORDER |
| 7 | 4 | Who is the most active issue creator? | select created_by__user_id, creator_name from issue group by creator_name order by count(*) desc limit 1 | GROUP,ORDER |

# Fine-tuning the T5 language model*

- T5 is an open-sourced multilingual Large Language Model; max. parameter size: 11B
- Optimised for transfer learning in the linguistic domain: every NLP task is converted into a **text-to-text format**



**Pretraining:**

"translate into German: that is good."

"Das ist gut."

"summarize: state authorities dispatched emergency crews tuesday to survey the damage after an onslaught of sever weather in mississipi..."

"Six people hospitalized after a storm in attala country."

**Fine-tuning:**

**"produce SQL: what are the issues for the week? DB schema: issue | id, name..."**

**"select id, name from issues where deadline < 2022-11-04"**

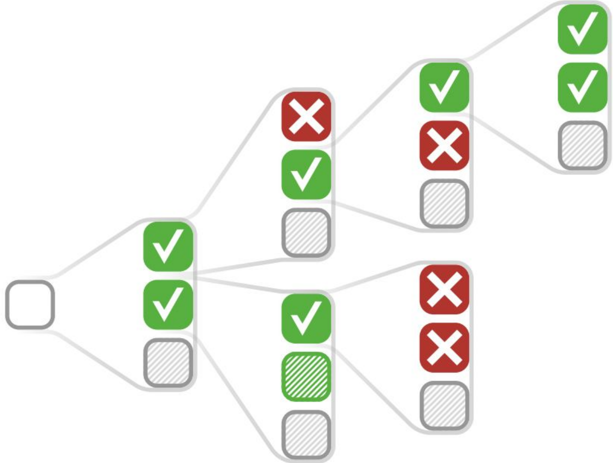
*Raffel et al. (2020): Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer

# Picard: Parsing Incrementally for Constrained Decoding*

“Bare” language models have an unconstrained output space;
no guarantee that the output is a well-formed SQL query

**Solution**: constrain decoder by rejecting unacceptable tokens at each time step:

- Keep only top-*k* highest probability tokens
- Reject tokens that fail checks



**3 constraint modes**:

- Lexing (SQL vocabulary)
- Parsing without guards (valid query syntax)
- Parsing with guards (valid syntax against DB schema)

**Benefits**:

- Ensures SQL validity & improves overall accuracy
- Not involved in pre-training/fine-tuning the model
- Prevents excessive use of world knowledge

*Tscholak et al. (2021): PICARD - Parsing Incrementally for Constrained Auto-Regressive Decoding from Language Models hellotwin.ai

your B2B SaaS superpower