Prototyp telemetrii ryzyka relacyjnego w narzędziach i chatbotach opartych na sztucznej inteligencji

Radosław Kołacki
radoslaw@kolacki.eu

Lipiec 2026

Streszczenie

Systemy konwersacyjne oparte na dużych modelach językowych są wdrażane jako pierwsza linia kontaktu w obsłudze klienta, jednak ich jakość jest dziś często mierzona głównie za pomocą metryk procesowych (czas odpowiedzi, wskaźnik defleksji, sentyment per wiadomość), które nie uchwytują zjawisk mogących wpływać na wynik relacyjny: braku postępu w dialogu, nieudanych prób naprawy błędów czy cichego odejścia klienta bez skargi.

Artykuł przedstawia **prototyp konceptualnych i implementacyjnych ram telemetrii ryzyka konwersacyjnego** (rozumianego jako obserwowalny operacyjnie wymiar szerszego ryzyka relacyjnego), oparty na dwóch elementach: **Operational Taxonomy** – taksonomii stanów i zdarzeń konwersacyjnych zaprojektowanej tak, by każda kategoria była uzasadnialna dowodem tekstowym i mapowała się na decyzję operacyjną, oraz **LLM-as-a-Judge** – zautomatyzowanym instrumencie pomiaru działającym na ustrukturyzowanych rubrykach z wymogiem evidence spans. System mierzy **dynamikę konwersacji** (jak rozmowa przebiegła), nie jakość agenta (czy odpowiedź była poprawna).

Prototyp działa w środowisku produkcyjnym i przetworzył dotychczas korpus obejmujący **1 146 sesji** (**8 374 wiadomości**) z pięciu głównych domen polskiego rynku (plus 60 sesji z domen dodatkowych). Walidacja empiryczna (inter-rater agreement, walidacja zależności z zewnętrznymi wskaźnikami rezultatu, tj. mierzalnymi rezultatami biznesowymi) jest w toku; plan walidacji opisano w rozdziale 7.

Uwaga metodologiczna: Prezentowane wyniki mają charakter wstępnych obserwacji z pilotażowego wdrożenia. Artykuł nie stanowi raportu z kontrolowanego badania walidacyjnego; jego celem jest przedstawienie koncepcji narzędzia, architektury prototypu oraz kierunków dalszej walidacji.

Słowa kluczowe: conversational AI, LLM-as-a-Judge, interaction risk, service recovery, emotion inference, relational telemetry, customer support automation

1. Wprowadzenie i motywacja

Rozmowy prowadzone przez systemy conversational AI stały się fragmentem infrastruktury wielu usług - od obsługi klienta i self-service po wsparcie zaawansowanych procesów biznesowych. W takim środowisku „jakość" rozmowy nie jest wyłącznie kwestią poprawności merytorycznej odpowiedzi. Równie istotne bywa to, czy użytkownik odczuwa postęp, czy rozumie kolejne kroki, oraz czy interakcja stabilizuje sytuację, czy przeciwnie - narasta w niej napięcie i nieufność. Z perspektywy organizacji są to zjawiska praktyczne: wpływają na eskalacje danej sytuacji, koszt obsługi, powtórne kontakty, a finalnie rezygnacje z dalszego procesu zakupu, wysokie wskaźniki rezygnacji (*churn*) czy reklamacje.

Wiele popularnych sposobów mierzenia „emocji" w tekście - np. sentyment liczony per wiadomość, proste klasyfikacje emocji, heurystyki słownikowe, a także ankiety po rozmowie - bywają przydatne, jednak w kontekście dialogu **mogą nie zawsze być wystarczające** do podjęcia decyzji operacyjnych. Wynika to nie z tego, że są „złe", lecz z tego, że dialog jest zjawiskiem sekwencyjnym: znaczenie wypowiedzi zależy od tego, co ją poprzedza, i od tego, czy rozmowa realnie posuwa sprawę do przodu. Część sygnałów, które w praktyce okazują się krytyczne, nie polega na doborze słów w jednej wiadomości, lecz na dynamice: powtórzeniach celu, braku postępu mimo kolejnych tur, reakcji na błąd, nagłych zmianach stylu wypowiedzi, albo na „uprzejmym zakończeniu" rozmowy, które niekoniecznie oznacza rozwiązanie problemu.

Niniejszy tekst wyrósł na bazie doświadczeń projektowych opartych na materiale obejmującym korpus 1 146 sesji konwersacyjnych (8 374 wiadomości). W takim materiale widać powtarzalne sytuacje, w których pojedyncza etykieta sentymentu lub prosta klasyfikacja emocji nie tłumaczy tego, co najbardziej interesuje zespół produktowy czy operacyjny: czy rozmowa zmierza ku rozwiązaniu, czy ryzyko eskalacji rośnie, czy użytkownik traci zaufanie, oraz w którym miejscu interakcji pojawia się „punkt zapalny". Nie chodzi więc o spór o definicję emocji, lecz o to, że użyteczna diagnoza w praktyce wymaga narzędzi, które są czułe na kontekst i przebieg rozmowy.

Z tego powodu przyjmujemy perspektywę węższą, ale bardziej kontrolowalną. Nie deklarujemy, że potrafimy ustalić „co użytkownik naprawdę czuje" w sensie psychologicznym. Interesuje nas raczej to, co da się uzasadnić w transkrypcie: **stany i ryzyka interakcji**, które przejawiają się w naturalny sposób w języku i strukturze dialogu, i które dają się powiązać z decyzjami operacyjnymi (np. kiedy eskalować do człowieka, jak projektować odpowiedzi naprawcze, jak monitorować spadek jakości). W kolejnych częściach omówimy zestaw kategorii takich stanów oraz hipotezy dotyczące sygnałów, po których można je przewidywać. Następnie opisujemy, jak skalować pomiar i anotację z użyciem podejścia typu **LLM-as-a-Judge** opartego na rubrykach i ustrukturyzowanych uzasadnieniach, tak aby wynik był nie tylko liczbowy, ale też możliwy do audytu.

1.1. Problem praktyczny

W obsłudze klienta emocja nie jest „tłem" rozmowy, lecz jednym z głównych mechanizmów decydujących o tym, czy interakcja zakończy się naprawą relacji, czy eskalacją konfliktu. Skala zjawiska jest przy tym empirycznie dobrze uchwytna: w badaniu amerykańskim (National Customer Rage Study, edycja 2025) **77%** respondentów deklaruje, że w ostatnich 12 miesiącach doświadczyło problemu z produktem lub usługą, a **64%** – że odczuwało z tego powodu „rage"; **50%** przyznaje, że podniosło głos, a kanały cyfrowe stały się dominującym medium skarg (**45%** vs **33%** telefon), przy jednoczesnym istotnym „publicznym" wektorze eskalacji (wpisy w social media) i deklarowanym braku reakcji firm (**43%** przypadków) (Broetzmann i Beinhacker, 2025). Dane te dotyczą rynku amerykańskiego i nie należy ich bezpośrednio przenosić na polski kontekst; przytaczamy je jako ilustrację skali zjawiska. Wskazują one na dwie cechy środowiska, w którym dziś działają systemy konwersacyjne:

1. wysoki poziom gotowości do eskalacji oraz
2. przeniesienie konfliktu do kanałów tekstowych, gdzie mikrosygnały emocji muszą być odczytywane bez wsparcia mowy ciała i prozodii.

Ten kontekst przecina się z praktyką automatyzacji obsługi: chatboty (w tym oparte na LLM) są wdrażane jako kosztowo skalowalne „pierwsze linie" kontaktu, ale raporty regulatorów pokazują, że ich skuteczność spada wraz ze złożonością problemu, a błędnie zaprojektowane ścieżki potrafią zamykać klientów w pętlach powtarzalnych, niepomocnych odpowiedzi („doom loops") bez sensownego wyjścia do człowieka (Consumer Financial Protection Bureau, 2023). Z perspektywy doświadczenia klienta nie jest to neutralny defekt użyteczności: to sytuacja, w której narzędzie obsługi staje się częścią problemu i podnosi koszt psychologiczny kontaktu.

W literaturze service recovery istnieje precyzyjne pojęcie opisujące ten mechanizm: **double deviation,** czyli drugi zawód po pierwotnej awarii, gdy firma nie potrafi adekwatnie naprawić błędu i sama próba naprawy staje się kolejną porażką (Smith i Bolton, 1998). Kluczowe jest to, że „druga porażka" nie działa addytywnie. Badania wskazują, że double deviation **intensyfikuje naruszenie zaufania** wywołane pierwszą awarią i wymaga innych strategii naprawczych niż pojedyncza „deviation" (w szczególności większej wagi przeprosin i obietnic nienawrotu vs. samej kompensacji finansowej) (Smith i Bolton, 1998). Jeśli więc system konwersacyjny ma pełnić rolę „front office", to jego błędy, zwłaszcza w eskalacji należy traktować nie jako zwykłe błędy klasyfikacji intencji, ale jako potencjalny generator drugiej porażki w logice relacji. Jednocześnie emocja, którą chcemy mierzyć, nie redukuje się do jednego wymiaru „pozytywne - negatywne". W pracach nad zachowaniami posprzedażowymi (post-sales) podkreśla się różnice między „dissatisfaction" a bardziej specyficzną emocją **anger** i jej związek z zachowaniami odwetowymi („get back") w usługach (Blodgett i in., 1997). To rozróżnienie jest praktycznie nośne: w obsłudze klienta interesuje nas nie tylko to, *czy* rozmowa jest negatywna, ale *czy* wchodzi w trajektorię eskalacji, gdzie rośnie **prawdopodobieństwo** działań kosztownych dla marki (odejście, publiczne skargi, odwet itd.).

W tym miejscu pojawia się robocza teza artykułu, sformułowana ostrożnie - nie jako rozstrzygnięcie psychologiczne, lecz jako hipoteza oparta na obserwacjach z wdrożeń i analizie rozmów: **klasyczne podejścia do „emocji w tekście" (np. proste sentymenty, słowniki emocji, kategorie podstawowe) mogą nie wystarczać do uchwycenia momentów przejścia w eskalację oraz do oceny jakości**

działań naprawczych w dialogu. W konsekwencji potrzebujemy operacjonalizacji emocji, która (a) jest użyteczna w kanałach tekstowych, (b) uwzględnia dynamikę rozmowy, oraz (c) da się mierzyć w sposób porównywalny i skalowalny. Narzędziem realizującym tę operacjonalizację jest **LLM-as-a-Judge** jako warstwa ewaluacyjna: silne modele mogą być wykorzystywane jako „sędziowie" do oceny jakości odpowiedzi i zgodności z kryteriami (np. deeskalacja, empatia, adekwatność naprawy). W benchmarkach porównawczych analizowanych przez Zheng i in. (2023) GPT-4 osiągał ponad 80% zgodności z preferencjami ludzi. Równolegle istnieją syntetyczne ujęcia tej praktyki i jej ograniczeń, co pozwala nadać podejściu solidną ramę analityczną (Liu i in., 2023).

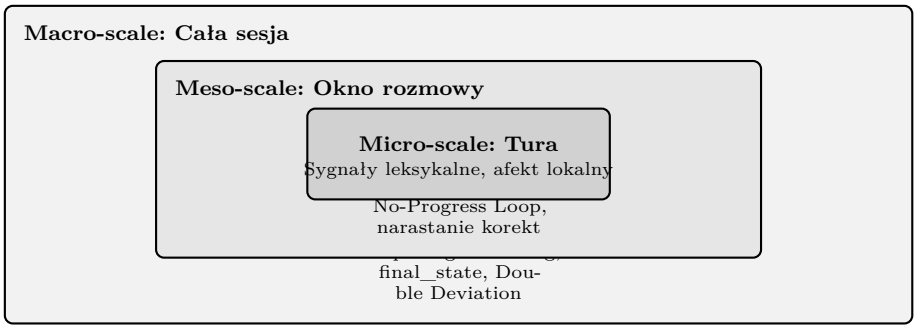
1.2. Emocja

Słowo „emocja" w artykule jest **skrótem myślowym** i nie aspiruje do bycia diagnozą psychologiczną. Interesuje nas nie tyle hipotetyczny stan wewnętrzny użytkownika, ile **mierzalny stan interakcji**, czyli taki, który można uzasadnić dowodami w tekście i przebiegu rozmowy. Z perspektywy teorii konstrukcji emocji, odchodzimy od poszukiwania uniwersalnych „odcisków palców" (fingerprints) ukrytych w mózgu czy ciele, ponieważ badania wykazują, że zmienność jest normą (Barrett, 2017). Zamiast tego skupiamy się na **stanach operacyjnych**: narastaniu napięcia, utracie poczucia sprawczości, przejściu w tryb eskalacji, wycofaniu (w tym „silent churn"), a także deeskalacji i powrocie do współpracy. Takie podejście pozwala uniknąć błędu, który w niniejszym artykule — inspirując się teorią konstrukcji emocji (Barrett, 2017) — określamy roboczo jako **mental inference fallacy** (sam termin został wprowadzony na potrzeby niniejszego artykułu). Zasada ta stanowi fundament Evidence-Only Judging, opisanej szczegółowo w rozdziale 4.1.

1.3. Jednostka analizy: tura / okno rozmowy / cała sesja

Analiza procesu obsługi wymaga monitorowania dynamiki zmian na trzech poziomach czasowych, co pozwala uchwycić emocję jako „film", a nie „zdjęcie":

1. **Tura (Micro-scale):** Analiza pojedynczego komunikatu pod kątem tekstowych sygnałów eskalacji i lokalnych wskaźników afektu, identyfikowanych na podstawie cech leksykalnych i pragmatycznych (Sanguinetti i in., 2020) oraz sekwencyjnych (Sandbank i in., 2018).
2. **Okno rozmowy (Meso-scale):** Kontekst kilku ostatnich tur, niezbędny do wykrycia zjawiska **No-Progress Loop** (pętli bez postępu), w której system powtarza nieskuteczne odpowiedzi, ignorując narastającą konfuzję klienta.
3. **Cała sesja (Macro-scale):** Perspektywa skumulowana, pozwalająca mierzyć skumulowaną jakość interakcji (Smith i Bolton, 1998). Na tym poziomie oceniamy końcowy efekt interakcji (`final_state`) w porównaniu do wejściowego (`initial_state`), co pozwala stwierdzić wystąpienie **Double Deviation** (Smith i Bolton, 1998, zob. rozdz. 1).



Rysunek 1: Trzy skale czasowe analizy konwersacji. Tura (micro) dostarcza sygnałów lokalnych; okno rozmowy (meso) pozwala wykryć pętle i narastanie tarcia; cała sesja (macro) umożliwia ocenę skumulowanego wyniku relacyjnego.

1.4. Założenia i granice uogólniania (domena, kanał, język, kontekst)

Choć mechanizmy konstruowania emocji są częścią ludzkiej biologii, ich interpretacja jest ściśle zależna od kontekstu społecznego i domenowego (Barrett, 2017).

- **Domena i Kontekst:** Reakcje klienta są osadzone w jego oczekiwaniach wobec danej usługi - to, co jest postrzegane jako sprawiedliwe w hotelu, może różnić się od oczekiwań w e-commerce (Blodgett i in., 1997).
- **Kanał:** Tekst jest uboższy o parametry akustyczne (np. wysokość tonu), które obiektywnie sygnalizują stres, dlatego modele oparte wyłącznie na tekście muszą uwzględniać tę lukę i nie powinny rościć sobie trafności porównywalnej z analizą multimodalną.
- **Język i Kultura:** Emocje nie są konstruktami uniwersalnymi - Hofstede i in. (2010) wykazują systematyczne różnice między kulturami w zakresie ekspresji emocjonalnej, norm grzeczności i stylów konfrontacji. To ma bezpośrednie konsekwencje operacyjne: to, co w jednej kulturze jest bezpośredną skargą, w innej może wyglądać jak neutralna prośba o informację. Bez uwzględnienia lokalnych norm komunikacyjnych system ryzykuje systematyczne błędne klasyfikacje, które prowadzą do zaniżenia priorytetu interwencji dla sesji wymagających pilnej reakcji (Barrett, 2017; Hofstede i in., 2010).
- **Ograniczenia zakresu:** Niniejszy artykuł koncentruje się na kanale tekstowym, rynku polskim i pięciu głównych domenach będących źródłem korpusu (plus 60 sesji z domen dodatkowych). Rozszerzenie na inne języki, domeny i kanały wymaga osobnej kalibracji rubryk i progowych parametrów. Przyjmujemy również, że sędzia AI ocenia interakcję według rygorystycznych rubryk jakościowych, a nie „wyczuwa" uczuć - co jest fundamentalnym warunkiem odporności systemu na obciążenia (*biases*) wbudowane w modele językowe.

2. Materiał empiryczny i kontekst

W realnych wdrożeniach **conversational AI** nie pracujemy na danych laboratoryjnych ani na zadaniach typu *one-shot classification*, tylko na transkryptach interakcji osadzonych w procesie biznesowym, z jego ograniczeniami (polityki, SLA, GDPR, ograniczona możliwość eskalacji, zmienność źródeł prawdy itp.). Z tego powodu rzetelny opis danych i kontekstu ich powstania staje się częścią metodologii, a nie dodatkiem redakcyjnym.

W literaturze ML/NLP rośnie nacisk na standaryzację dokumentowania danych i warunków ich użycia: zarówno w postaci **„datasheets"** (motywacja, skład, proces pozyskania, zastosowania i ryzyka), jak i **„data statements"** (charakterystyka populacji, języka, kanału i kontekstu wytwarzania tekstu), właśnie po to, by ograniczyć nadinterpretacje i fałszywe uogólnienia. W tym duchu traktujemy nasz materiał jako **korpus obserwacyjny**: jego wartość polega na tym, że ujawnia typowe mechanizmy eskalacji i naprawy w warunkach produkcyjnych, ale jednocześnie wymusza ostrożność w wnioskowaniu poza domenę, kanał i język.

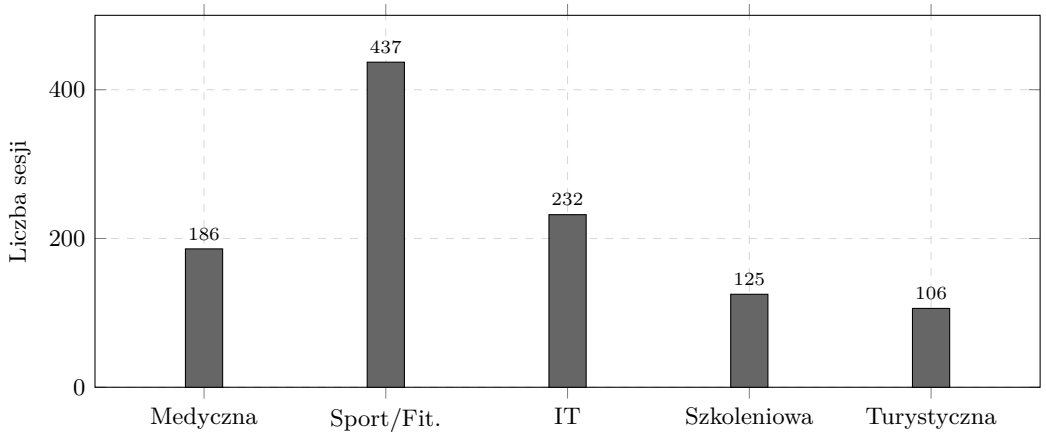
2.1. Charakterystyka korpusu: transkrypty i dynamika polskiego rynku usług

Punktem wyjścia dla budowy systemu jest autentyczny materiał z warunków produkcyjnych, a nie dane laboratoryjne - co ma kluczowe znaczenie dla trafności rubryk i kalibracji progowych. Materiał badawczy stanowi korpus **8 374 wiadomości** pogrupowanych w **1 146 sesjach** konwersacyjnych, pochodzących od klientów w Polsce z takich branż jak medycyna, sport (trenerzy fitness), centrum pomocy w aplikacjach mobilnych oraz chaty sprzedażowe na stronach internetowych B2B (business-to-business). Po zastosowaniu filtrów jakości do szczegółowej analizy zakwalifikowano **1 146 sesji** – tę liczbę prezentuje Tabela 1. Kryteria włączenia: (1) sesja zawiera minimum dwie wiadomości (co najmniej jedną turę użytkownika i jedną turę asystenta); (2) sesja jest oznaczona flagą `mvp_v2_complete`, co oznacza, że pipeline scoringowy przetworzył sesję bez błędów i zwrócił kompletny output JSON ze wszystkimi polami schematu (tab. 7). Sesje odrzucone z powodu niepełnego outputu (np. timeout API, błąd parsowania JSON) nie wchodzą do zbioru analitycznego. Pełny przepływ filtrowania przedstawia się następująco: (1) pula wyjściowa: wszystkie sesje zarejestrowane w systemie; (2) odrzucenie sesji z jedną wiadomością (mniej niż dwie wiadomości); (3) odrzucenie sesji z błędnym lub niepełnym outputem scoringowym (`scoring_failed` lub brakujące pola JSON; $n = 0$ w obecnym zbiorze); wynik: **1 146 sesji analitycznych**. Liczba sesji z jedną wiadomością odrzuconych w kroku (2) nie jest rejestrowana w obecnej wersji pipeline'u i zostanie uzupełniona w kolejnej iteracji.

Istotnym założeniem metodologicznym jest fakt, że **sesje składające się z tylko jednej wiadomości nie podlegają analizie**, ponieważ każda rozmowa jest inicjowana przez Asystenta AI. W tym modelu sesja jednoelementowa oznacza jedynie rozpoczęcie interakcji przez bota bez reakcji ze strony klienta, co nie pozwala na analizę trajektorii relacyjnej. Sesje takie mogą jednak stanowić odrębną klasę zdarzeń związaną z porzuceniem interakcji i powinny być analizowane osobno w przyszłych iteracjach. W konsekwencji

Metryka	**Medycz.**	**Sport/Fit.**	**IT**	**Szkol.**	**Turyst.**	**CAŁOŚĆ**
Liczba sesji	186	437	232	125	106	**1 146**
Mediana wiad./sesję	5,0	5,0	3,0	4,0	5,0	**5,0**
Średnio wiad./sesję	6,3	8,4	6,3	6,7	7,4	**7,3**
P90 wiad./sesję	12	17	13	12	12	**13**
% effort = 3	90,9%	92,2%	85,3%	96,0%	77,4%	**86,8%**
% z pętlą	1,1%	0,5%	0,4%	0,0%	0,0%	**0,4%**
% z exit intent	0,0%	7,6%	1,7%	0,0%	0,0%	**3,2%**
Rozkład P0/P1/P2/P3	0/12/4/84%	8/7/5/80%	1/18/4/77%	0/13/6/81%	0/2/6/92%	**3/10/5/82%**

Tabela 1: Statystyki opisowe korpusu (aktualizacja: 15 lipca 2026). Dane z systemu produkcyjnego ($n = 1\,146$ sesji). Skala effort: 1–5 (wartości 4 i 5 nie wystąpiły w obecnym datasecie). Kolumny: Medycyna (186 sesji), Sport/Fitness (437), IT (232), Szkoleniowa (125), Turystyczna (106). Całość obejmuje także pozostałe domeny (60 sesji).



Rysunek 2: Rozkład sesji w korpusie według domeny ($n = 1\,146$). Domena Sport/Fitness stanowi największy segment (38,1%), co odzwierciedla specyfikę bazy klientów. Suma pięciu nazwanych domen wynosi 1 086; pozostałe 60 sesji pochodzi z domen o niskiej liczebności.

faktyczna analiza sędziego AI rozpoczyna się od **drugiej wiadomości w sesji** - pierwszej odpowiedzi klienta.

Na etapie projektowania taksonomii, przed uruchomieniem obecnej wersji automatycznego pipeline'u (potoku przetwarzania), w jakościowym przeglądzie rozmów zaobserwowano kilka powtarzalnych wzorców. Najczęściej rozpoznawanym zdarzeniem był **No-Progress Loop**, identyfikowany w większości domen, choć z odmiennymi markerami językowymi: w sektorze medycznym objawiał się głównie przez neutralne reformułowania tego samego pytania o procedurę, podczas gdy w e-commerce – przez wyraźne skracanie wiadomości i rosnącą bezpośredniość żądań. Wzorzec **Justice Gap** był najsilniej widoczny w sesjach związanych z zarządzaniem kontem i subskrypcjami, szczególnie gdy system odmawiał bez wyjaśnienia procedury – co koresponduje z ustaleniami Blodgetta, Hilla i Taxa o priorytecie sprawiedliwości proceduralnej nad dystrybutywną (Blodgett i in., 1997). Wzorzec **Failed Recovery** obserwowano w domenie wsparcia technicznego, gdzie system wielokrotnie proponował kroki diagnostyczne już wykonane przez użytkownika. *Uwaga:* powyższe obserwacje pochodzą z ręcznej analizy jakościowej przeprowadzonej na etapie projektowania rubryk, nie z wyników automatycznego sędziego. W bieżącym pipeline'u pole `failed_recovery` wykazuje zerową wariancję (sekcja 7.1), co sugeruje problem z detekcją automatyczną, nie brak zjawiska. Obserwacje te mają charakter indukcyjny – wpłynęły na kształt taksonomii i dobór progów opisanych w rozdziale 4, nie są jednak statystycznie reprezentatywne i wymagają formalnej walidacji. Skupienie na pierwszej reakcji użytkownika jest kluczowe z kilku powodów. Architektura opisana w artykule działa w trybie **offline** (post-hoc scoring transkryptów), zaprojektowanym z myślą o możliwości rozszerzenia do trybu real-time w kolejnej iteracji.

- **Prewencja „Silent Churn" (Cichego odejścia):** Brak deeskalacji już w pierwszym kroku często skutkuje trwałym wycofaniem klienta z relacji - bez żadnej jawnej skargi, która mogłaby uruchomić interwencję (Technical Assistance Research Programs [TARP], 1986). (Ustalenia TARP pochodzą z 1986 roku; przeniesienie mechanizmu „cichego odejścia" na współczesne kanały cyfrowe, gdzie bariery zmiany dostawcy są niższe, jest hipotezą wymagającą osobnej walidacji). To zjawisko jest szczególnie kosztowne biznesowo, bo nie pozostawia żadnego sygnału w klasycznych wskaźnikach reklamacyjnych.
- **Wczesna detekcja ryzyka relacyjnego:** Hipoteza projektowa inspirowana teorią Barrett (2017) zakłada, że w branżach takich jak medycyna niepewność i brak jasnych wyjaśnień mogą być silniejszymi wyzwalaczami negatywnego afektu niż sam czas oczekiwania. Różne domeny wymagają więc różnych progowych parametrów dla tych samych kategorii taksonomicznych.
- **Unikanie Double Deviation (Podwójnego odchylenia):** Ocena post-hoc pierwszych kroków naprawy błędu pozwala wykryć wzorce, w których nieudana próba naprawy może nasilać negatywną ocenę pierwotnej awarii (Smith i Bolton, 1998). Smith i Bolton (1998) wykazali, że sposób przeprowadzenia recovery wpływa na ocenę całego doświadczenia, a zawiedzione oczekiwanie korekty może być oceniane surowiej niż sam pierwotny błąd.

Na podstawie obserwacji z tak zróżnicowanego materiału, od wsparcia technicznego po administracyjną obsługę placówek medycznych, rubryki sędziego zostały zaprojektowane tak, aby uwzględniać domenowy charakter sygnałów: ta sama kategoria emocjonalna (np. złość) w różnych branżach stanowi unikalną „instancję", a nie stały odcisk palca (Barrett, 2017). Sędzia interpretuje daną kategorię w kontekście domeny na podstawie instrukcji zawartych w rubryce, nie w wyniku fine-tuningu na korpusie. Taka architektura ma wspierać generowanie ustrukturyzowanej **telemetrii relacyjnej** już od pierwszego kontaktu klienta z systemem, co może umożliwić wcześniejszą priorytetyzację sesji potencjalnie związanych z ryzykiem odejścia (TARP, 1986).

2.2. Przygotowanie danych i protokół anotacji: od transkryptu do materiału pomiarowego

Jeżeli emocje mają być mierzone w sposób porównywalny, to transkrypt rozmowy musi zostać potraktowany jak dane o określonej strukturze, a nie jak „opis sytuacji". W praktyce oznacza to, że jeszcze przed właściwą ewaluacją potrzebujemy dwóch warstw porządku:

- po pierwsze - **porządku dokumentacyjnego**, który jasno opisuje, skąd wzięły się dane i do czego wolno ich używać;
- po drugie - **porządku operacyjnego**, który zamienia rozmowę na jednostki obserwacji (tury/okna/sesje) oraz na zestaw zmiennych, które da się później agregować i testować.

W części dokumentacyjnej inspirujemy się praktykami wypracowanymi w ML/NLP: „datasheets for datasets" jako standard opisu motywacji, składu, pozyskania i ryzyk zbioru danych, oraz „data statements" jako forma ujawnienia warunków językowych i populacyjnych, od których zależy możliwość generalizacji. Te ramy są narzędziem metodologicznym: bez nich bardzo łatwo przypisać zachowaniu językowemu

użytkownika status „uniwersalnego sygnału emocji", podczas gdy w rzeczywistości jest ono efektem kanału, domeny, presji czasu albo norm stylu. W analogiczny sposób, na poziomie modeli oceniających, przydatna jest logika „model cards": jawne określenie przeznaczenia, ograniczeń i warunków testów, zamiast milczącego założenia, że miara „działa zawsze". Warstwa operacyjna zaczyna się od ujednolicenia transkryptów do wspólnego formatu: role (użytkownik/system), kolejność tur, podstawowe metadane (kanał, ewentualny typ sprawy) i znaczników czasu. Dopiero na takim materiale można budować „obserwacje", które nie są opisem jakości w stylu eseju, lecz **wypełnieniem ustrukturyzowanych rubryk oceny jakościowej, które mapują dynamikę interakcji na konkretne wartości numeryczne i flagi logiczne**. Kluczowym elementem protokołu anotacji jest dekonstrukcja każdej sesji przez pryzmat trzech wymiarów sprawiedliwości:

- **dystrybutywną** (czy wynik jest postrzegany jako uczciwy),
- **proceduralną** (czy proces był sprawny i przejrzysty),
- **interakcyjną** (czy system potraktował klienta z empatią i szacunkiem), co pozwala na przewidywanie przyszłych zachowań klienta (Blodgett i in., 1997; Tax i in., 1998).

Protokół ten rygorystycznie wymusza zasadę Evidence-Only Judging (zob. rozdz. 4.1), nakazując sędziemu AI identyfikację stanów operacyjnych wyłącznie na podstawie dowodów obecnych w tekście (Barrett, 2017). Ostatecznym produktem tego procesu jest **struktura danych JSON**, zawierająca zautomatyzowany wskaźnik **Priority Score (PS)**, który w wersji MVP priorytetyzuje eskalację danej sprawy wyłącznie na podstawie sygnałów konwersacyjnych (docelowo planowane jest rozszerzenie o metadane biznesowe, takie jak wartość klienta CLV czy priorytety SLA – zob. sekcja 3.3). Takie przygotowanie strukturalne stanowi podstawę dla precyzyjnej analizy sędziego AI już od pierwszej odpowiedzi użytkownika w sesji.

2.3. Reprezentatywność korpusu: obciążenia selekcji i zróżnicowanie typów interakcji

Najważniejsze ryzyko metodologiczne w pracy z danymi serwisowymi wynika z faktu, że **transkrypty rozmów nie stanowią reprezentatywnej próby z ogólnej populacji użytkowników**, lecz są wynikiem silnie nielosowego procesu selekcji (Gebru i in., 2021). Transkrypty obejmują przede wszystkim sytuacje, w których użytkownicy podejmują interakcję w związku z trudnością, niejasnością lub potrzebą, której nie mogą samodzielnie zrealizować (TARP, 1986; Broetzmann i Beinhacker, 2025). Należy jednak podkreślić, że **mechanizm selekcji nie ogranicza korpusu wyłącznie do interakcji o skrajnym ładunku emocjonalnym**. Analiza 8 374 wiadomości pozwala na wyróżnienie kilku typów interakcji, z których każdy niesie odmienne ryzyko dla relacji z użytkownikiem:

- **Zapytania informacyjne i produktowe (Service/Product Inquiries):** Pytania typu „ile kosztuje produkt?" czy „jak dobrać plan?" nie są jedynie prośbami o dane. W ujęciu predykcyjnego przetwarzania (Clark, 2016) każda niejasność w odpowiedzi może generować błąd przewidywania, który obciąża zasoby poznawcze użytkownika.
- **Zarządzanie kontem i subskrypcjami (Transactional/Administrative):** Prośby o rezygnację z subskrypcji, zwrot pieniędzy czy zmianę pakietu są krytycznymi testami dla **sprawiedliwości proceduralnej** (Blodgett i in., 1997; Tax i in., 1998). Jeśli system utrudnia te procesy lub nie rozpoznaje intencji, afekt klienta skupia się wokół poczucia niesprawiedliwości proceduralnej, które w badaniach zachowań poskargowych było istotnie związane z intencją ponownego zakupu i negatywnym *word-of-mouth* (Blodgett i in., 1997; Tax i in., 1998; TARP, 1986).
- **Usability i wsparcie nawigacyjne (Usability/Navigation):** Pytania „gdzie znajdę zrealizowane treningi?" wskazują na tarcie w realizacji celów użytkownika. Brak natychmiastowej gratyfikacji informacyjnej może powodować narastanie negatywnej walencji. Sędzia AI musi wychwycić moment, w którym drobna niedogodność zmienia się w frustrację operacyjną - a to przejście często jest niewidoczne w klasycznej analizie sentymentu, bo język użytkownika pozostaje nadal neutralny.
- **Wsparcie techniczne i awarie (Tech Support/Service Failure):** Zgłoszenia błędów w aplikacji to klasyczne sytuacje **Service Failure** (Smith i Bolton, 1998). Badania nad *service recovery* sugerują, że sposób reakcji organizacji na zgłoszony błąd może mieć istotny wpływ na późniejszą ocenę doświadczenia i deklarowaną lojalność (Smith i Bolton, 1998). Nieudana próba naprawy prowadzi do zjawiska **Double Deviation**, mogąc nasilać negatywną ocenę i afekt wywołane pierwotną awarią (Smith i Bolton, 1998; Bougie i in., 2003). W tej niszy może skalować się również ryzyko zachowań odwetowych, w tym negatywnego WOM (Bougie i in., 2003).
- **Potrzeby motywacyjne i afektywne (Personal Support):** Komunikaty typu „chce mi się spać, potrzebuję motywacji!" można interpretować – w ujęciu teorii konstrukcji emocji – jako raport o **stanie afektu** (niska walencja, niskie pobudzenie) i wyczerpanym budżecie ciała użytkownika (Barrett, 2017).

W tej niszy rola systemu AI przesuwa się z czystej informacji w stronę aktywnej regulacji emocjonalnej poprzez odpowiednie sygnały komunikacyjne.

Podsumowując, mimo że korpus jest „przefiltrowany" przez potrzebę kontaktu, zawiera on zróżnicowany przekrój intencji występujących w korpusie. Rubryki sędziego zostały zaprojektowane tak, aby uwzględniać tę różnorodność: „rezygnacja z subskrypcji" może być procesem chłodnym i technicznym, podczas gdy „pytanie o 4 jajka w diecie" może nieść silny ładunek niepewności o własne cele zdrowotne.

3. Prace powiązane

Podejście zaproponowane w artykule lokuje się na przecięciu trzech dojrzałych, lecz wciąż dynamicznie ewoluujących nurtów badawczych, z których każdy wnosi istotne narzędzia metodologiczne, ale jednocześnie posiada specyficzne ograniczenia w kontekście operacyjnej obsługi klienta. Pierwszym z nich jest **klasyczna analiza sentymentu i opinii**, która przez dekady koncentrowała się głównie na ujęciach binarnych lub prostych skalach polaryzacji pozytywnej i negatywnej. Tradycyjne metody, oparte na modelach leksykonowych czy algorytmach takich jak SVM (ang. Support Vector Machine), operują zazwyczaj na poziomie pojedynczego dokumentu lub zdania, co w realiach serwisowych uniemożliwia zrozumienie ewolucji afektu klienta w odpowiedzi na kolejne komunikaty systemu. Choć współczesne badania nad afektem podkreślają, że walencja (przyjemność/nieprzyjemność) towarzyszy człowiekowi przez całe życie, samo określenie sentymentu jest jedynie „częściowym obrazem" stanu emocjonalnego. W kontekście serwisowym podejście to często pomija proces pełnej rekonstrukcji dynamiki dialogu, co uniemożliwia skuteczne wykrywanie zjawisk takich jak **Double Deviation** (zob. rozdz. 1).

Drugi nurt to **analiza emocji (emotion analysis) w NLP**, dziedzina rosnąca niezwykle szybko, ale wciąż pozbawiona stabilnego konsensusu metodologicznego co do zakresu stosowanych ram teoretycznych. Badacze wciąż wahają się między klasycznymi modelami dyskretnych emocji podstawowych Ekmana a bardziej złożonymi systemami, takimi jak koło emocji Plutchika, co prowadzi do fragmentacji terminologicznej. Kluczowym wyzwaniem pozostaje tu uwzględnienie **kontekstu kulturowego i demograficznego;** jak wskazuje teoria konstrukcji, bez rygorystycznego oddzielenia formy tekstu od hipotetycznego stanu wewnętrznego, systemy te są narażone na błąd **mental inference fallacy** (Barrett, 2017). Warto zauważyć, że argument ten jest samozwrotny: jeśli percepcja emocji u innych jest aktem kategoryzacji, nie detekcji, to również sędzia LLM - oceniając transkrypt - konstruuje emocje na bazie swoich danych treningowych i promptu, nie „odczytuje" ich z tekstu. Ta samozwrotność stanowi dodatkowe uzasadnienie dla zasady Evidence-Only Judging, opisanej w rozdziale 4. Ponadto standardowe podejścia NLP rzadko uwzględniają zjawisko **granularności emocjonalnej**, czyli zdolności do konstruowania precyzyjnych i zróżnicowanych stanów (np. odróżnienia dejekcji od irytacji), co ma bezpośredni wpływ na trafność reakcji systemu w sytuacjach kryzysowych (Barrett, 2017).

Trzeci nurt, stanowiący bezpośredni kontekst dla sędziego AI opisanego w artykule, dotyczy **ewaluacji systemów dialogowych przy użyciu silnych modeli językowych (LLM-as-a-Judge)**. W warunkach benchmarkowych raportuje się wysoką zgodność takich modeli z preferencjami ludzkimi, co czyni je atrakcyjnym narzędziem do automatycznej oceny jakości konwersacji. Niemniej jednak literatura przedmiotu coraz częściej opisuje typowe źródła błędów i obciążeń w takich ocenach, wynikające m.in. z faktu, że modele te przejmują uprzedzenia demograficzne i lingwistyczne ze zbiorów treningowych, co może prowadzić do systematycznej niesprawiedliwości, np. błędnego oceniania specyficznego stylu komunikacji jako agresywnego. Proponowane podejście integruje te trzy nurty w ramach systemu telemetrii relacyjnej: taksonomia operacyjna dostarcza kategorii ryzyka, LLM-as-a-Judge dostarcza skalowalnego instrumentu pomiaru, a zestaw KPI zamienia wyniki pomiaru w decyzje operacyjne. W ten sposób sędzia AI staje się ogniwem łączącym analizę tekstu z operacyjną potrzebą ochrony lojalności klienta i zapobiegania zjawisku **silent churn**.

3.1. Analiza sentymentu i emocji w NLP: możliwości i ograniczenia

W tradycji **sentiment/opinion mining** „emocja" bywa najczęściej redukowana do prostego wymiaru afektywnego nastawienia: polaryzacji dodatniej lub ujemnej, ewentualnie uzupełnionej o stopień natężenia. Takie podejście jest pragmatyczne i dobrze sprawdza się w zadaniach o niskiej dynamice, jak agregacja opinii w recenzjach produktów, jednak wykazuje krytyczne ograniczenia w analizie rozmów serwisowych (Smith i Bolton, 1998). W interakcjach z systemami konwersacyjnymi **ton wypowiedzi bywa wtórny wobec braku postępu w dialogu**; użytkownik może zachowywać neutralność językową, podczas gdy narastający

brak postępu buduje frustrację poza poziomem tekstu. Warto odnotować, że istnieją podejścia sekwencyjne (m.in. modele LSTM i BERT aplikowane do sekwencji tur), które częściowo adresują problem dynamiki; ich ograniczeniem pozostaje jednak skupienie na klasyfikacji emocji lub sentymentu *per wypowiedź*, a nie na ocenie jakości naprawy relacji jako całości sesji. Osobny nurt stanowią metryki automatycznej ewaluacji dialogu, takie jak USR (*Unsupervised and Reference-free*) czy FED (*Fine-grained Evaluation of Dialogue*), które oceniają jakość rozmowy bez konieczności posiadania referencyjnych odpowiedzi (Mehri i Eskenazi, 2020a,b). Równolegle seria wyzwań DBDC (*Dialogue Breakdown Detection Challenge*) zaproponowała formalne ramy detekcji momentów, w których dialog przestaje funkcjonować (Higashinaka i in., 2016). Podejścia te są bliższe proponowanej telemetrii niż klasyczny sentyment, lecz koncentrują się na jakości generacji (czy odpowiedź jest koherentna, trafna, interesująca), nie na dynamice relacji (czy interakcja naprawia problem i chroni zaufanie) - co stanowi odmienną perspektywę analityczną. Co więcej, proces eskalacji nie musi manifestować się stałym „negatywnym sentymentem" w każdej turze - narastanie frustracji bywa niewidoczne w pojedynczych wiadomościach, a ujawnia się dopiero w dynamice sekwencji. Literatura przedmiotu wskazuje, że to nie pierwotna awaria usługi (*service failure*), lecz nieudolna próba naprawy - **Double Deviation** (Smith i Bolton, 1998) - generuje najbardziej destrukcyjne stany afektywne.

Rozwijająca się w obszarze NLP **emotion analysis** idzie o krok dalej, oferując klasyfikacje konkretnych stanów (np. złość, strach, smutek), ale jednocześnie zmaga się z brakiem ujednoliconej terminologii i stabilnego konsensusu co do ram teoretycznych (Buechel i Hahn, 2016). W najnowszych przeglądach pola podkreśla się niejasność w zakresie tego, co dokładnie jest mierzone: czy jest to stan mentalny autora, emocja wywołana u odbiorcy, czy może emocja jako obiektywna cecha tekstu (Barrett, 2017). Z perspektywy teorii konstrukcji każda kategoria emocjonalna (np. „Złość") jest w rzeczywistości populacją wysoce zróżnicowanych instancji, co oznacza, że złość na błąd w aplikacji mobilnej będzie miała zupełnie inne markery lingwistyczne niż złość na brak terminu u lekarza (Barrett, 2017). Brak rygorystycznego rozdziału formy tekstu od hipotetycznego stanu wewnętrznego sprawia, że modele NLP często wpadają w pułapkę błędu **mental inference fallacy** (Barrett, 2017, zob. rozdz. 1). W konsekwencji, nawet gdy model „rozpoznaje emocje", wynik pozostaje trudno porównywalny między domenami: to, co w sektorze fitness jest uznawane za motywacyjne pobudzenie, w sektorze medycznym może być klasyfikowane jako agresywna niecierpliwość (Barrett, 2017; Hofstede i in., 2010).

Proponowana telemetria relacyjna stara się wypełnić tę lukę, wychodząc poza katalog etykiet w stronę analizy **sprawiedliwości proceduralnej i interakcyjnej** (Blodgett i in., 1997; Tax i in., 1998). System musi rozumieć, że celem użytkownika jest nie tyle wyrażenie uczuć, co realizacja konkretnego zadania, a każda niejasność w odpowiedzi systemu generuje koszt poznawczy i erozję zaufania – którą sędzia AI musi zarejestrować zanim dojdzie do potencjalnej utraty lojalności.

3.2. Sygnały konwersacyjne: dynamika dialogu, eskalacje i naprawa błędów

Jeżeli w niniejszym artykule traktujemy „emocje" jako mierzalny stan interakcji, to naturalnym punktem zaczepienia nie są deklaracje typu „jestem zły", tylko sygnały, że rozmowa przestaje działać: rośnie koszt poznawczy użytkownika, spada postęp zadaniowy, a system nie potrafi przejść od błędu do naprawy. W literaturze o **dialogue breakdown** taka sytuacja bywa definiowana jako moment, w którym uczestnik nie jest w stanie sensownie kontynuować rozmowy w danym kierunku (Higashinaka i in., 2016). Ocena, czy doszło do „zerwania" dialogu, wymaga miar opartych na obserwowalnych markerach sekwencyjnych - nie na wnioskach o stanach mentalnych - aby uniknąć błędu **mental inference fallacy** (zob. rozdz. 1). W praktyce analitycznej wygodnie jest operacjonalizować te zjawiska jako sygnały konwersacyjne na trzech poziomach:

1. **Dynamika dialogu (czy rozmowa „idzie do przodu").** Najbardziej stabilne empirycznie sygnały mają charakter sekwencyjny: powtórzenia i przeformułowania po stronie użytkownika, „ping-pong" tych samych intencji oraz rosnąca długość tur bez rozwiązania problemu. Każda taka tura generuje kumulujący się koszt poznawczy (Clark, 2016). W danych produkcyjnych to „narastanie korekt" (gdy użytkownik doprecyzowuje i poprawia system) często poprzedza frustrację nawet wtedy, gdy w samym tekście nie ma jawnie negatywnych słów, co sprawia, że klasyczne modele sentymentu mogą „nie widzieć problemu" (Smith i Bolton, 1998).
2. **Eskalacje.** Eskalacja jest sygnałem krytycznym, w którym użytkownik przechodzi od prób naprawy dialogu do jawnego żądania alternatywnego kanału lub połączenia z człowiekiem. Nieudana eskalacja w pierwszym kroku może skutkować zjawiskiem **Silent Churn** (cichego odejścia), potencjalnie osłabiając lojalność klienta (TARP, 1986). Sędzia AI wykorzystuje te sygnały do obliczenia wskaźnika **Priority**

Score (PS), który syntetyzuje wykryty afekt z ryzykiem biznesowym naruszenia zasad sprawiedliwości interakcyjnej i proceduralnej (Blodgett i in., 1997; Tax i in., 1998).

3. **„Failure recovery" (czy system umie wyjść z błędu).** Zdolność do naprawy błędu (*Service Recovery*) jest kluczowym momentem prawdy w relacji (Smith i Bolton, 1998). Jeżeli bot nie potrafi skutecznie zareagować na zgłoszony problem (np. błąd techniczny lub niejasną ofertę), dochodzi do **Double Deviation** (Smith i Bolton, 1998). Z drugiej strony, sprawnie przeprowadzone wyjście z błędu może w niektórych przypadkach doprowadzić do zjawiska określanego jako **Service Recovery Paradox** - sytuacji, w której jakość obsługi kryzysu skutkuje wyższą deklarowaną lojalnością niż przed wystąpieniem błędu. Zjawisko to jest jednak kontrowersyjne empirycznie i nie występuje powszechnie (Smith i Bolton, 1998). Sędzia AI monitoruje ten proces, mierząc, czy system dostarcza odpowiednie wyjaśnienia i empatię, co w literaturze service recovery bywa wskazywane jako istotny element skutecznej naprawy relacji (Smith i Bolton, 1998).

3.3. LLM-as-a-Judge: skalowalna ewaluacja rozmów i kontrola obciążeń

Współczesny zwrot ku metodologii **LLM-as-a-Judge** (LLM jako sędzia) wynika z konieczności znalezienia skalowalnej i efektywnej kosztowo alternatywy dla ocen ludzkich w procesie ewaluacji systemów generatywnych. W benchmarkach porównawczych analizowanych przez Zheng i in. (2023) GPT-4 osiągał ponad 80% zgodności z preferencjami ludzi, co czyni modele językowe obiecującym narzędziem w analizie dużych zbiorów danych konwersacyjnych. Należy jednak pamiętać, że modele te nie są neutralnymi obserwatorami i wykazują specyficzne błędy poznawcze, takie jak **position bias** (uprzywilejowanie odpowiedzi na określonej pozycji w prompcie) czy **verbosity bias** (skłonność do wyższego oceniania wypowiedzi dłuższych i bardziej „gadatliwych"). Dodatkowym wyzwaniem jest zjawisko **self-enhancement bias**, w którym model wykazuje tendencję do faworyzowania tekstów o strukturze i stylu zbliżonym do jego własnych generacji. Aby mitygować te ryzyka, Liu i in. (2023) proponują w metodologii **G-Eval** stosowanie ustrukturyzowanych **rubryk oceniania** oraz jawnych kroków wnioskowania typu **Chain-of-Thought**. Takie podejście wymusza na modelu sędziowskim rozbicie złożonego afektu klienta na mniejsze, mierzalne składowe i oparcie werdyktu na konkretnych **dowodach tekstowych** (tzw. spanach tekstu). Oceny mogą przyjmować formę **pointwise** (ocena jednostkowa), **pairwise** (porównywanie par wypowiedzi) lub **listwise** (rankingowanie całych list), przy czym każda z tych konfiguracji wymaga rygorystycznej kalibracji w celu uniknięcia nadpewności modelu (Liu i in., 2023). Równolegle rozwijają się wyspecjalizowane modele sędziowskie (np. JudgeLM (Zhu i in., 2023), Auto-J (Li i in., 2024)), fine-tunowane na danych z ocenami ludzkimi. Wyspecjalizowany sędzia może osiągać wyższą zgodność z ludźmi, lecz rodzi to ryzyko overfittingu do zbioru treningowego. W kontekście omawianym w artykule, sędzia AI nie jest traktowany jako „wyrocznia" posiadająca wgląd w duszę użytkownika, lecz jako precyzyjny **komponent telemetrii relacyjnej** (Zheng i in., 2023; Liu i in., 2023). Kluczowe jest uniknięcie błędu **mental inference fallacy** (zob. rozdz. 1). Zamiast tego sędzia AI analizuje **dynamikę interakcji**, mierząc m.in. narastanie korekt po stronie użytkownika czy tarcie operacyjne. Tak zaprojektowany system oblicza wskaźnik **Priority Score (PS)**, który w obecnej wersji MVP opiera się wyłącznie na sygnałach konwersacyjnych (P0–P3). Docelowa architektura zakłada rozszerzenie PS o metadane biznesowe (wartość klienta CLV, priorytety SLA), jednak ta funkcjonalność nie została jeszcze zaimplementowana i wymaga osobnej kalibracji. Sędzia AI jest zaprojektowany jako narzędzie **wczesnego ostrzegania**, które ma wspierać identyfikację sesji zmierzających ku **Double Deviation** (Smith i Bolton, 1998). Celem systemu jest wsparcie wcześniejszej identyfikacji sesji potencjalnie związanych z ryzykiem utraty lojalności, co może umożliwić skuteczniejszą **naprawę serwisową** (*service recovery*) (Smith i Bolton, 1998; TARP, 1986).

3.4. Wnioski z przeglądu: gdzie powstaje luka w zastosowaniach operacyjnych

Przegląd prac powiązanych w niniejszym artykule prowadzi do dość spójnego wniosku: istnieją zaawansowane narzędzia do mierzenia nastroju i opinii w tekście oraz stale rosnący korpus badań nad emocjami w NLP, jednak w konkretnych zastosowaniach serwisowych ich użyteczność operacyjna często załamuje się w dwóch kluczowych obszarach. Po pierwsze, wynika to z braku jawnej definicji tego, co dokładnie jest mierzone w trakcie trwania dialogu - czy jest to stan mentalny autora, afekt „zawarty" w samej wypowiedzi, dynamiczny stan relacji, czy może bezpośrednie ryzyko eskalacji, co często prowadzi do błędu **mental inference fallacy** (Barrett, 2017). Po drugie, problemem jest niedopasowanie jednostki analizy, gdzie systemy skupiają się na pojedynczej wypowiedzi, ignorując sekwencyjną naturę i dynamikę rozmowy, co uniemożliwia dostrzeżenie zjawisk takich jak narastające tarcie poznawcze wywołane brakiem postępu w rozwiązaniu problemu (Barrett, 2017; Dixon i in., 2013). Proces eskalacji nie musi manifestować się

stałym negatywnym sentymentem w każdej turze - narastanie frustracji bywa niewidoczne w pojedynczych wiadomościach, a ujawnia się dopiero w dynamice sekwencji - czego podejście per-wiadomość z definicji nie jest w stanie uchwycić. Równolegle literatura HCI dotycząca **breakdown & repair** precyzyjnie opisuje procesy awarii i naprawy dialogu, ale rzadziej dostarcza „telemetrii gotowej do bezpośredniego wdrożenia", czyli zestawu stanów i zdarzeń, które można by konsekwentnie oznaczać na dużej liczbie rozmów i mapować na konkretne decyzje produktowe, takie jak inteligentny routing czy polityka **service recovery** (Smith i Bolton, 1998). Z perspektywy operacyjnej wyzwaniem nie jest więc brak teorii, lecz brak funkcjonalnego mostu między koncepcjami teoretycznymi a mierzalnym procesem, który pozwoliłby uniknąć **Double Deviation** (Smith i Bolton, 1998).

Wreszcie, podejście **LLM-as-a-Judge** otwiera obiecującą ścieżkę skalowania ewaluacji, lecz same prace o sędziach AI podkreślają, że bez rygorystycznych rubryk, kontroli obciążeń (takich jak **position bias** czy **verbosity bias**) oraz procedur stabilności łatwo wprowadzić nowy rodzaj niejawnej zmienności. Powoduje to, że wynik może zostać zdyskwalifikowany jako wiarygodna metryka zarządcza, jeśli nie da się go obronić w procesie audytu, co dowodzi, że sama zdolność mierzenia w skali nie jest tożsama z mierzeniem odpowiedzialnym i porównywalnym. W kolejnym rozdziale przejdziemy do szczegółowego opisu zaproponowanej metodyki, która integruje te trzy nurty w spójny system wczesnego ostrzegania przed eskalacją.

3.5. Kontekst rynkowy i krajobraz narzędziowy

Proponowany framework operuje na styku czterech ekosystemów narzędziowych (szczegółowe omówienie: Aneks E):

(1) Biblioteki NLP (VADER, BERT, HerBERT (Mroczkowski i in., 2021), XLM-RoBERTa (Conneau i in., 2020)) – operują na poziomie pojedynczej wypowiedzi; nie modelują dynamiki sesji ani pętli konwersacyjnych. Modele transformerowe (Devlin i in., 2019; Liu i in., 2019; He i in., 2021) stanowią istotny postęp w klasyfikacji emocji (Demszky i in., 2020), lecz typowy klasyfikator turn-level nie jest projektowany do wykrywania zjawisk sekwencyjnych. **(2) Frameworki ewaluacji LLM** (DeepEval, RAGAS, TruLens) – mierzą poprawność i bezpieczeństwo wyniku (faithfulness, halucynacje, toksyczność), nie dynamikę relacji. **(3) Platformy observability** (Langfuse, LangSmith) – monitorują latencję, koszty i regresje promptów; celują w inżynierów ML, nie zespoły CX. **(4) Platformy conversation analytics i QA** – najbliżsi funkcjonalni odpowiednicy: Zendesk QA, Microsoft Dynamics 365 Quality Scoring, Observe.AI. Są jednak zamknięte w ekosystemach swoich dostawców i w publicznie dostępnej dokumentacji nie oferują taksonomii opartej na literaturze service recovery.

W przeglądzie publicznie udokumentowanych funkcji wybranych platform (lipiec 2026 r.; przegląd niesystematyczny) nie zidentyfikowano narzędzia łączącego: standalone API agnostyczne platformowo, scorowanie transkryptu konwersacji AI pod kątem dynamiki relacyjnej (pętle, nieudane naprawy, sprawiedliwość proceduralna), strukturę wyjściową JSON z evidence spans. Proponowany framework ma na celu wypełnienie tej luki.

Kierunek rozwoju: fine-tunowane transformery. Obecna implementacja frameworku wykorzystuje LLM ogólnego przeznaczenia jako sędziego. Kolejnym planowanym etapem jest fine-tuning wyspecjalizowanego modelu transformerowego (np. na bazie HerBERT (Rybak i in., 2020; Mroczkowski i in., 2021)) na danych z systemu produkcyjnego, z trzema celami:

1. redukcja kosztu i latencji inferencji (znacznie mniejszy model domenowy zamiast zewnętrznego modelu ogólnego przeznaczenia),
2. poprawa kalibracji kulturowej i językowej (model trenowany na polskich transkryptach obsługowych, nie na ogólnym korpusie angielskim),
3. deterministyczność i powtarzalność wyników.

W literaturze NLP obserwuje się, że dodatkowy pretraining na korpusie domenowym przed fine-tuningiem na zadanie docelowe może poprawiać wyniki w zadaniach specjalistycznych — hipoteza ta wymaga jednak osobnej weryfikacji w kontekście polskojęzycznej obsługi klienta. Podejście to jest zbieżne z wynikami badań nad fine-tunowanymi modelami sędziowskimi (JudgeLM (Zhu i in., 2023), Auto-J (Li i in., 2024)). Na obecnym etapie system działa z sędzią LLM ogólnego przeznaczenia; porównanie z wersją fine-tunowaną jest zaplanowane jako jeden z kierunków dalszych badań.

Wymiar	**Sentyment NLP**	**Ewaluacja LLM**	**Observ-ability**	**Conv. Intelligence (sales)**	**QA platforms**	**Niniejszy framework**
Jednostka analizy	Wypowiedź	Zapytanie	Trace	Rozmowa sprzedażowa	Interakcja agent-klient	Sesja konwersacyjna AI
Co mierzy	Polaryzację tekstu	Poprawność, safety	Latencję, koszty	Deal signals, obiekcje	Compliance, ton, kompletność	Dynamikę relacji, naprawę, zaufanie
Wykrywa pętle	Nie	Nie	Częściowo	Nie	Częściowo	Tak
Wykrywa eskalację	Nie	Nie	Nie	Częściowo	Częściowo	Tak (STATE → fazowa logika)
Wykrywa nieudaną naprawę	Nie	Nie	Nie	Nie	Nie	Tak* (Failed Recovery)
Standalone API	Tak (biblioteki)	Tak (open-source)	Tak/Nie	Nie (SaaS)	Nie (SaaS)	Tak
Taksonomia service recovery	Nie	Nie	Nie	Nie	Nie	Tak
Grupa docelowa	Data scientists	ML engineers	ML ops	Sales leaders	QA managers	Zespoły CX / product

* W obecnym zbiorze pilotażowym failed_recovery ma zerową wariancję (wszystkie sesje = false); detekcja jest zaimplementowana w rubryce, lecz nie została jeszcze empirycznie przetestowana na przypadkach pozytywnych.

Tabela 2: Porównanie kategorii narzędzi conversation analytics.

4. Taksonomia stanów i ryzyk rozmowy (Operational Taxonomy)

Proponowana taksonomia ma charakter celowo „operacyjny": kategorie są dobrane tak, by:

1. dało się je rozpoznać w transkrypcie bez spekulacji o wnętrzu użytkownika,
2. miały sens w dynamice dialogu (mogą rosnąć, spadać, przechodzić jedna w drugą), oraz
3. mapowały się na decyzje produktowe lub procesowe (np. eskalacja, zmiana strategii naprawy, priorytetyzacja błędów).

Taki wybór jest spójny z podejściem, które zamiast „nazewnictwa emocji" traktuje je jako zmienne operacyjne powiązane z zachowaniem i wynikiem interakcji. W ujęciu Barrett (2017) emocje nie są dyskretnymi „odciskami palców" w mózgu, lecz konstruowanymi kategoriami, co oznacza, że próba ich jednoznacznej klasyfikacji na podstawie samego tekstu jest z natury obarczona ryzykiem nadinterpretacji. Dlatego taksonomia celowo operuje na poziomie obserwowalnych markerów lingwistycznych i strukturalnych, a nie na poziomie domniemanych stanów wewnętrznych użytkownika. Takie ujęcie jest również spójne z praktyką inżynierii NLP: przeglądy taksonomii emocjonalnych stosowanych w analizie sentymentu wskazują, że systemy oparte na grubszych, behawioralnie zakotwiczonych kategoriach osiągają wyższą zgodność anotatorów niż systemy opierające się na subtelnych rozróżnieniach afektywnych (Mohammad i Turney, 2013; Buechel i Hahn, 2016).

4.1. Kryteria projektowe taksonomii

Konstrukcja taksonomii opisanej w artykule opiera się na kryteriach projektowych, które mają na celu przekształcenie teoretycznych ram neuronauki afektywnej w konkretne narzędzie telemetrii relacyjnej. Pierwszym i fundamentalnym wymogiem jest **audytowalność kategorii**, co oznacza, że każdy werdykt wydany przez sędziego AI lub anotatora ludzkiego **musi być poparty konkretnym materiałem dowodowym** w postaci fragmentów (spanów) tekstu. Wynika to z konieczności mitygowania błędu **mental inference fallacy** (Barrett, 2017, zob. rozdz. 1). W systemach wysokiej stawki, jakimi są platformy obsługi klienta, brak możliwości wskazania lingwistycznego źródła oceny uniemożliwia rzetelny audyt jakości oraz blokuje proces ciągłego doskonalenia algorytmów. Wymóg audytowalności jest spójny z zasadami przejrzystości, dokumentacji i nadzoru ludzkiego zawartymi w Rozporządzeniu (UE) 2024/1689 (AI Act), o ile konkretny kontekst wdrożenia podlega jego przepisom (European Parliament and Council

of the European Union, 2024). Drugim kryterium jest zachowanie odpowiedniej **grubości (granularności) kategorii**, tak aby taksonomia była odporna na rozbieżności interpretacyjne między różnymi domenami biznesowymi. Przeglądy literatury NLP wskazują, że nadmierne rozdrobnienie etykiet emocjonalnych prowadzi do drastycznego spadku zgodności anotatorów, ponieważ granice między subtelnymi stanami są płynne i subiektywne (Mohammad i Turney, 2013; Buechel i Hahn, 2016). Zastosowanie ujęcia opartego na *population thinking* pozwala traktować np. „Złość" nie jako jeden konkretny odcisk palca, lecz jako populację różnorodnych instancji językowych, co zwiększa stabilność oznaczeń i pozwala modelowi na skuteczniejszą generalizację. Podejście to jest zbieżne z konstruktywistycznym modelem emocji (Barrett, 2017), w którym kategorie emocjonalne mają charakter statystyczny, a nie esencjalny - każda instancja „złości" może wyglądać inaczej na poziomie lingwistycznym, fizjologicznym i behawioralnym. Trzeci filar projektowy to wyraźne **rozdzielenie stanów od zdarzeń**, co ma kluczowe znaczenie operacyjne. **Stany** opisują aktualną dynamikę relacji, manifestującą się np. jako narastające tarcie poznawcze, utrata poczucia sprawczości lub narastająca nieufność wobec systemu. Z kolei **zdarzenia** to dyskretne, obiektywnie wykrywalne fakty konwersacyjne, takie jak wpadnięcie bota w pętlę, nieudana próba naprawy błędu czy jawna prośba o przełączenie do człowieka. Taki podział jest praktyczny, ponieważ zdarzenia można łatwo identyfikować metodami regułowymi (regex, pattern matching, heurystyki sekwencyjne), podczas gdy stany są wyliczane jako funkcja sekwencji tych zdarzeń w czasie. To rozróżnienie nawiązuje do klasycznej dystynkcji w modelowaniu procesów między zmiennymi stanu (*state variables*) a zdarzeniami wyzwalającymi (*trigger events*), stosowanej w modelowaniu systemów dynamicznych. Czwartym kryterium jest **kompatybilność z mierzeniem ciągłym**, co pozwala uniknąć ograniczeń tradycyjnych podejść typu „per wiadomość". Taksonomia musi operować na poziomie tury, okna konwersacyjnego lub całej sesji, aby uchwycić procesy takie jak **Double Deviation** (Smith i Bolton, 1998; Joireman i in., 2013). Uwzględnienie dynamiki dialogu pozwala sędziemu AI na monitorowanie **pętli predykcyjnych** i wykrywanie momentów, w których błąd przewidywania systemu staje się dla użytkownika zbyt kosztowny metabolicznie, co bezpośrednio przekłada się na wskaźnik **Priority Score**. W ujęciu predykcyjnego przetwarzania (*predictive processing*) każda nieudana tura generuje sygnał błędu predykcji, który kumuluje się i obniża gotowość użytkownika do dalszego angażowania zasobów poznawczych w interakcję (Clark, 2016). Mierzenie ciągłe pozwala uchwycić tę kumulację, czego podejście „per wiadomość" z definicji nie jest w stanie zrobić. Wdrożenie tych czterech kryteriów stanowi niezbędny fundament dla kolejnego etapu prac, którym jest zdefiniowanie operacyjne poszczególnych kategorii taksonomii w celu ich automatycznej detekcji. Należy podkreślić, że proponowana taksonomia ma charakter **indukcyjny i roboczy** - została wyprowadzona z obserwacji korpusu produkcyjnego i wymaga formalnej walidacji zewnętrznej (inter-rater agreement, korelacja z outcome) przed traktowaniem jej kategorii jako ostatecznych.

Typ	Kategoria	Definicja skrócona	PS
STATE	Low-Progress Friction	Narastający koszt poznawczy bez postępu w dialogu	P2→P1
STATE	High-Arousal Threshold	Przekroczenie progu intensywności emocjonalnej	P0/P1
STATE	Justice Gap	Naruszenie sprawiedliwości proceduralnej / interakcyjnej	P1*
EVENT	No-Progress Loop	Powtarzalność intencji bez przyrostu informacji ($\geq$ 2 cykle)	P1
EVENT	Failed Recovery (Double Deviation)	Nieudana próba naprawy po błędzie	P1
EVENT	Breakdown / Exit Intent	Jawna lub funkcjonalna rezygnacja z kanału	P1 (P0)
METRIC	Priority Score	Syntetyczny wskaźnik ryzyka (P0–P3)	–
GOV	Evidence-Only Judging	Wymóg uzasadnienia werdyktu dowodami tekstowymi	–

Tabela 3: Taksonomia operacyjna ryzyk relacyjnych – wersja skrócona. Pełna tabela z uzasadnieniami teoretycznymi i celami operacyjnymi: Aneks, Tabela 6. *Justice Gap (P1) nie jest w obecnym MVP zaimplementowany jako bezpośredni override – manifestuje się pośrednio przez inne wyzwalacze (zob. sekcja 4.2).

4.2. Formalna definicja Priority Score (wersja MVP)

Priority Score (PS) jest zmienną porządkową o czterech poziomach (P0–P3), gdzie P0 oznacza najwyższy priorytet interwencji. W wersji MVP priorytetyzacja opiera się wyłącznie na sygnałach konwersacyjnych - rozszerzenie o metadane biznesowe (CLV, SLA) jest zaplanowane w kolejnej iteracji. **Reguła przypisania:** PS = max(severity) spośród wykrytych kategorii. System rejestruje dwa warianty: `peak_priority_score` (najwyższy PS osiągnięty w trakcie sesji) oraz `closing_priority_score` (PS po uwzględnieniu ewentualnego udanego recovery – jeśli turning point jest pozytywny i intencja została zrealizowana, closing PS może być niższy niż peak). Tabela decyzyjna przypisania peak PS:

- **P0 (natychmiastowa interwencja):** legal_threat = true LUB (High-Arousal Threshold przekroczony ORAZ exit_intent = hard).
- **P1 (priorytetowy przegląd):** Dowolne z: loop_count $\geq$ 2, failed_recovery = true, exit_intent $\in$ {soft, hard}, effort_score $\geq$ 4. *Uwaga dot. Justice Gap:* w obecnym MVP Justice Gap nie posiada osobnego pola w schemacie JSON – manifestuje się pośrednio przez effort_score, trajektorię walencji i opis turning_point. Dedykowane pole `justice_gap: {procedural, interactional, distributive}` jest zaplanowane w kolejnej iteracji (zob. sekcja 6.4). Do tego czasu sesje z sygnałami Justice Gap mogą uzyskać P1 przez inne wyzwalacze (effort_score $\geq$ 4, exit_intent $\in$ {soft, hard}), nie przez bezpośredni override.
- **P2 (monitorowanie):** Dowolne z: Low-Progress Friction bez eskalacji, effort_score = 3, loop_count = 1.
- **P3 (rutyna):** Brak wykrytych sygnałów ryzyka.

Powyższa tabela jest kompletna – nie wymaga osobnych reguł nadpisujących, ponieważ wszystkie progi eskalacji są już zawarte w definicjach poszczególnych poziomów. **Reguła closing PS:** Closing PS może zostać obniżony względem peak PS o **co najwyżej jeden band** (np. P2→P3, P1→P2), pod warunkiem łącznego spełnienia trzech kryteriów: (1) turning point jest pozytywny, (2) intent_fulfillment = true, albo intent_fulfillment = partial z fulfillment_quality = helpful_partial (wartości deflected i apparently_misinformed blokują obniżenie), (3) po zakończeniu recovery nie pozostaje aktywny żaden warunek P0 ani P1. Warunki legal_threat oraz exit_intent = hard nie podlegają automatycznemu obniżeniu – nawet przy udanym recovery closing PS nie może spaść poniżej P1 w tych przypadkach. Reguła jest deterministyczna: sędzia LLM nie decyduje o obniżeniu samodzielnie, lecz stosuje powyższe warunki.

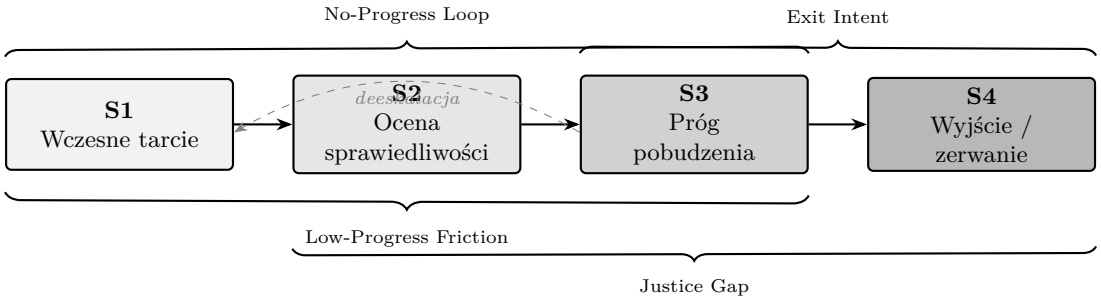
Uwaga: Progi (effort $\geq$ 4, loop_count $\geq$ 2) mają charakter heurystyczny i wymagają kalibracji empirycznej per kategoria intencji po zebraniu danych outcome (zob. rozdz. 6.5).

4.3. Logika fazowa taksonomii (S1→S4)

Fazy oznaczone w tabeli (S1–S4) nie są sztywnymi etapami, lecz opisują typową trajektorię eskalacji ryzyka relacyjnego w rozmowie obsługowej. Ich sekwencja odpowiada narastaniu kosztu interakcji po stronie użytkownika:

- **S1 (wczesne tarcie):** Użytkownik ma problem i szuka rozwiązania. Język jest jeszcze neutralny lub lekko sfrustrowany. Kluczowe jest wykrycie Low-Progress Friction zanim przejdzie w jawną eskalację.
- **S2 (ocena sprawiedliwości):** Użytkownik zaczyna oceniać nie tylko *co* dostaje, ale *jak* jest traktowany. Tu aktywuje się Justice Gap - naruszenia proceduralne i interakcyjne mogą być ważniejsze niż sam brak rozwiązania (Blodgett i in., 1997; Tax i in., 1998).
- **S3 (próg pobudzenia):** Intensywność przekracza próg, po którym dalsza automatyzacja jest kontrproduktywna. High-Arousal Threshold oznacza, że koszt każdej kolejnej nieadekwatnej tury rośnie nieliniowo.
- **S4 (wyjście/zerwanie):** Użytkownik komunikuje (jawnie lub funkcjonalnie) rezygnację z kanału. Breakdown/Exit Intent to punkt, po którym odzyskanie relacji jest znacząco trudniejsze i droższe.

Ważne jest, że przejścia między fazami nie są deterministyczne: rozmowa może „skoczyć" z S1 do S3 (np. po odkryciu przez użytkownika, że bot podał fałszywą informację) lub cofnąć się z S3 do S1 (skuteczna deeskalacja). Taksonomia zakłada monitorowanie ciągłe, nie jednorazową klasyfikację.



Rysunek 3: Trajektoria ryzyka relacyjnego w fazach S1–S4 z nałożonymi zakresami kategorii taksonomii. Przejścia nie są deterministyczne - skuteczna deeskalacja może cofnąć rozmowę z S3 do S1.

5. LLM-as-a-Judge jako instrument pomiaru jakości konwersacji

Wdrożenia modeli językowych w obsłudze klienta zwykle zaczynają się od elementu najbardziej widocznego: „zróbmy chatbota". To zrozumiałe, bo interfejs rozmowy daje szybki efekt produktu. Jednak w praktyce operacyjnej największa dźwignia pojawia się często gdzie indziej: w momencie, gdy model nie bierze udziału w interakcji, tylko zostaje użyty jako **instrument pomiarowy**, oceniający transkrypty rozmów po fakcie według ustalonych kryteriów. Ten paradygmat można nazwać roboczo *LLM-as-a-Judge*: model pełni rolę sędziego, który nie „współprowadzi rozmowy", lecz **standaryzuje ocenę jakości naprawy** i pozwala zamienić tekst w dane o stałej strukturze (telemetrię).

Opisany framework został zaimplementowany jako system produkcyjny działający w architekturze offline. System scoruje sesje w trybie ciągłym, przetwarzając transkrypty z pięciu głównych domen (medyczna, sport i fitness, IT/technologia, szkoleniowa, turystyczna) oraz domen dodatkowych o niskiej liczebności. Wyniki scoringu są dostępne przez API. Plan walidacji empirycznej opisany jest w rozdziale 7. Tabela 4 zawiera minimalne informacje o konfiguracji systemu.

W wersji opisanej w tym artykule sędzia działa w trybie **offline**: scoring odbywa się po zakończeniu sesji, na pełnych transkryptach. To świadomy wybór architektoniczny - tryb offline jest prostszy do wdrożenia, tani w eksploatacji i wystarczający do większości zastosowań analitycznych i szkoleniowych. Jednocześnie cały schemat danych (JSON output, rubryki, progi) został zaprojektowany tak, by można go było bez przepisywania przenieść do trybu real-time - różnicą byłoby jedynie uruchomienie sędziego na żywo po każdej turze zamiast po zakończeniu sesji. W literaturze ewaluacji modeli podejście to zostało szeroko opisane w kontekście oceniania odpowiedzi innych modeli: od MT-Bench i Chatbot Arena (Zheng i in., 2023) po nowsze ramy oceny, takie jak G-Eval (Liu i in., 2023), które pokazują zarówno potencjał, jak i typowe błędy sędziów (m.in. bias pozycji, rozwlekłości, auto-faworyzowania). W naszym kontekście

Parametr	**Wartość / opis**
Model sędziowski	GPT-4o (OpenAI); identyfikator API: `gpt-4o-2024-08-06`
Temperatura	0,0 (konfiguracja minimalizująca losowość; nie gwarantuje pełnej deterministyczności backendu API)
Język promptu	angielski (transkrypty polskojęzyczne)
Format outputu	JSON Schema z wymuszeniem struktury (`response_format: json_schema`)
Retry policy	do 3 prób przy niepoprawnym JSON; sesje bez poprawnego wyniku oznaczane jako `scoring_failed` ($n = 0$ w obecnym korpusie)
Okno kontekstu	pełny transkrypt sesji (do 128k tokenów)
Opening / core / closing	liczone wyłącznie z tur użytkownika (nie asystenta): opening = pierwsza tura użytkownika; closing = ostatnia tura użytkownika; core = tury użytkownika 2. . . (n−1). Sesje z dokładnie 2 turami: core jest pusty, Δ = closing − opening. Sesje z 1 turą: opening = closing, Δ := NA (nie 0); traktowane jako sesje bez interpretowalnej trajektorii
Detekcja pętli	pre-pass regułowy; `loop_count` = liczba cykli użytkownik–bot, w których użytkownik powtarza lub przeformułowuje wcześniejszą intencję (w oknie 4 tur) ORAZ odpowiedź systemu nie dostarcza nowej, użytecznej informacji ani postępu zadaniowego. Samo powtórzenie intencji nie jest wystarczające – jeśli system między powtórzeniami dostarczył nową informację, cykl nie jest liczony. `loop_detected` := true gdy loop_count ≥ 1
Wersja promptu / rubryk	prompt v1.2, rubric v1.2 (lipiec 2025, scoring: czerwiec–lipiec 2026); codebook dostępny na życzenie; prompt nie jest publikowany ze względu na charakter własności intelektualnej; hash wersji promptu (SHA-256) udostępniany na życzenie w celu weryfikacji reprodukowalności
Anonimizacja	usunięcie PII (imiona, numery telefonów, adresy e-mail) przed scoringiem
Data scoringu	czerwiec–lipiec 2026
Spójność pipeline'u	wszystkie 1 146 sesji przetworzono tą samą wersją promptu, rubryk i modelu

Tabela 4: Parametry implementacyjne prototypu sędziego AI.

(reklamacje, wsparcie, onboarding, procesy zakupowe) stawka jest inna: nie chodzi o „jakość generacji" w sensie lingwistycznym, tylko o to, czy interakcja **naprawiła problem i nie uszkodziła relacji**. Dlatego „sędzia" nie ma być generatorem kolejnych narracji o emocjach. Ma być elementem infrastruktury pomiarowej: tak ustawionym, by dało się go audytować, wersjonować i kalibrować.

Krytyczna granica zakresu. System opisany w tym artykule to **telemetria jakości konwersacji dla celów triage'u** - nie ewaluacja agenta. Różnica jest fundamentalna i wyznacza granice wnioskowania dostępne z samego transkryptu. Ocena dynamiki konwersacji (jak rozmowa przebiegła) jest możliwa z transkryptu. Ocena jakości agenta (czy agent udzielił *poprawnej* odpowiedzi) wymaga czegoś, czego transkrypt nie zawiera: wiedzy o tym, jaka odpowiedź była poprawna, co agent był w stanie zrobić, i czego powinien był się trzymać zgodnie z polityką. Bez dostępu do tych trzech źródeł - ground truth, capability boundaries, policy documents - sygnały klienta mogą być reakcją na działanie agenta, ale równie dobrze na nierozwiązywalny problem, nierealistyczne oczekiwania lub czynniki zewnętrzne. Sygnał klienta jest konieczny, ale nie wystarczający do ewaluacji agenta.

System jest mocny tam, gdzie chce być mocny: w wykrywaniu sygnałów wymagających interwencji i priorytetyzowaniu sesji do przeglądu. Budowanie na tej podstawie ewaluacji jakości agenta wymaga kolejnego kroku - połączenia telemetrii konwersacyjnej z danymi o poprawności odpowiedzi - co jest naturalną ewolucją frameworku, nie jego wadą. Warto podkreślić, że podejście LLM-as-a-Judge w kontekście obsługi klienta różni się od jego zastosowań w benchmarkach modeli pod dwoma istotnymi względami. Po pierwsze, w benchmarkach (MT-Bench, Chatbot Arena) oceniana jest jakość pojedynczej odpowiedzi, podczas gdy w obsłudze klienta kluczowa jest **trajektoria całej rozmowy** - sekwencja tur, dynamika napięcia, kumulacja błędów. Po drugie, w benchmarkach „złota odpowiedź" jest zwykle znana lub możliwa do ustalenia; w obsłudze klienta „dobra naprawa" zależy od kontekstu biznesowego, polityki firmy, historii klienta i specyfiki problemu. Dlatego sędzia w naszym modelu operuje na rubrykach wielowymiarowych, a nie na prostym porównaniu z referencją.

5.1. Przesunięcie przedmiotu oceny: od detekcji emocji do ewaluacji naprawy relacji

Kluczowa zmiana metodologiczna polega na przesunięciu pytania:

- z: „czy klient był zły?"
- na: „czy rozmowa była skuteczną naprawą - merytorycznie, proceduralnie i relacyjnie?"

To rozróżnienie jest praktyczne. W wielu procesach obsługi „negatywność" bywa konsekwencją samego faktu kontaktu (reklamacja, opóźnienie, awaria), ale o **wyniku biznesowym** decyduje trajektoria: czy rozmowa obniżyła napięcie, przywróciła sprawczość i domknęła intencję, czy też pogorszyła sytuację i uruchomiła mechanizmy eskalacji oraz odejścia „bez hałasu". Literatura service recovery od lat pokazuje, że szkoda nie wynika wyłącznie z pierwotnego błędu usługi. Krytycznym predyktorem jest **jakość próby naprawy** - zwłaszcza wtedy, gdy próba naprawy zawodzi i dokłada klientowi kolejne straty (zaufania, czasu, godności) (Smith i Bolton, 1998; Joireman i in., 2013). Dlatego w LLM-as-a-Judge mierzymy przede wszystkim **jakość naprawy**, a emocje traktujemy jako sygnały funkcjonalne: informują o ryzyku, o tarciu procesu, o naruszeniu sprawiedliwości w komunikacji, o spadku zaufania do informacji. Przesunięcie to ma również konsekwencje dla tego, jakie dane uznajemy za wartościowe. Klasyczny sentiment analysis traktuje każdą wiadomość jako niezależną jednostkę analizy i przypisuje jej walencję. W naszym podejściu jednostką analizy jest **sesja** (lub jej fazy: opening/core/closing), a walencja jest jedynie jedną z wielu zmiennych wejściowych - obok kompletności rozwiązania, wiarygodności proceduralnej, dynamiki pętli i punktu zwrotnego. To zmiana z *per-message sentiment* na *per-session repair telemetry.*

5.2. Zasada audytowalności oceny: rubryki oparte na dowodach tekstowych

Najbardziej kosztowny błąd w tradycyjnym QA nie polega na tym, że ocena jest „subiektywna". Problem polega na tym, że ocena często próbuje rozstrzygać o **stanach wewnętrznych** klienta, których transkrypt nie dowodzi. Inspirując się teorią konstrukcji emocji (Barrett, 2017), wprowadzamy dla tej skłonności termin roboczy *mental inference fallacy*: tendencję do wnioskowania o cudzych stanach mentalnych bez wystarczającej podstawy w danych. W modelu sędziego tę pułapkę redukujemy przez zasadę: **Oceniamy zachowania i właściwości interakcji, nie „wnętrze" użytkownika.** Praktycznie oznacza to rubryki, które:

1. mają **jednoznaczne kryteria**,
2. są powiązane z **ryzykiem operacyjnym**,
3. wymagają wskazania **dowodu w tekście** (spans, cytowane fragmenty), gdy wynik ma być audytowalny.

To podejście ma dodatkową zaletę: ujednolica standardy między światem H2H (człowiek–człowiek) i Bot2H (bot–człowiek). W obu przypadkach oceniamy tę samą jednostkę: przebieg interakcji i jej skutki. Dzięki temu możliwe jest porównanie jakości obsługi między kanałami na wspólnej skali, co jest warunkiem koniecznym racjonalnej alokacji ruchu między botem a agentem ludzkim. Wymóg evidence-based oceny jest również zbieżny z najlepszymi praktykami w projektowaniu rubryk ewaluacyjnych: stosowanie jawnych kryteriów i analitycznych rubryk może poprawiać przejrzystość oraz spójność ocen (Jonsson i Svingby, 2007).

5.3. Wymiary oceny: zestaw rubryk ewaluacyjnych

Poniższy zestaw nie jest „pełną taksonomią emocji", a jedynie szkic rubryk, które da się skalować i które mapują się na decyzje operacyjne.

5.3.1. Poprawność merytoryczna i zgodność z polityką

[Warstwa rozszerzona – wymaga zewnętrznych źródeł prawdy: bazy wiedzy, regulaminów, polityk.] Czy odpowiedź była zgodna z aktualną bazą wiedzy, regulaminem, polityką zwrotów, SLA? To fundament: błąd merytoryczny nie jest „pomyłką w zdaniu", tylko generatorem utraty przewidywalności po stronie klienta. W ujęciu predykcyjnego przetwarzania (Clark, 2016) błąd merytoryczny może obniżać zaufanie do całego kanału, nie tylko do konkretnej odpowiedzi. W obecnym MVP, operującym wyłącznie na transkrypcie, sędzia nie ma dostępu do bazy wiedzy ani polityk – detekcja błędów merytorycznych ogranicza się do przypadków, w których sprzeczność jest jawna w samym transkrypcie (np. bot podaje dwie sprzeczne informacje w tej samej sesji). Pełna weryfikacja poprawności wymaga integracji ze źródłami prawdy i stanowi planowane rozszerzenie.

5.3.2. Wiarygodność proceduralna (detekcja jawnych obietnic)

Czy system złożył jawną, kategoryczną obietnicę dotyczącą wyniku, terminu lub działania („na pewno dziś", „na pewno natychmiast")? W service recovery fałszywa obietnica jest silnym zapalnikiem eskalacji, bo tworzy drugi, bardziej dotkliwy zawód (Smith i Bolton, 1998; Joireman i in., 2013). Obietnica bez pokrycia jest operacyjnie groźniejsza niż brak obietnicy, ponieważ aktywuje oczekiwanie, którego zawód jest następnie atrybucyjnie przypisywany niekompetencji lub nieuczciwości firmy (Joireman i in., 2013). *[Warstwa rozszerzona – stwierdzenie, czy obietnica jest „bez pokrycia", wymaga zewnętrznego źródła prawdy (SLA, stany magazynowe, harmonogramy).]* W obecnym MVP pole `promise_detected` ogranicza się do detekcji jawnej obietnicy w transkrypcie (tj. sędzia oznacza fakt złożenia obietnicy, nie weryfikuje jej spełnialności). Pełna walidacja wymaga integracji z systemami backendowymi i stanowi rozszerzenie warstwy zewnętrznej.

5.3.3. Kompletność rozwiązania (transcript-inferred intent fulfillment)

Czy rozmowa domknęła intencję użytkownika – nie tylko formalnie „zamknęła ticket"? W praktyce intencja bywa wielowątkowa (wynik + wyjaśnienie + zabezpieczenie na przyszłość). To rubryka ściśle operacyjna: jej brak wraca w recontact/reopen. Wskaźnik ten pełni funkcję zbliżoną do operacyjnego FCR (*First Contact Resolution*), lecz jest wnioskowany wyłącznie z transkryptu — sędzia ocenia *pozór* domknięcia, a nie faktyczne rozwiązanie w systemach backendowych. Wartość pola `intent_fulfillment` jest wnioskowana wyłącznie z transkryptu (*transcript-inferred*): sędzia ocenia, czy na podstawie przebiegu rozmowy intencja *wydaje się* domknięta, a nie czy faktycznie została zrealizowana w systemach backendowych. Rozróżnienie to jest istotne, ponieważ bot może deklarować rozwiązanie problemu, które nie zostało faktycznie wykonane.

5.3.4. Sprawiedliwość interakcyjna i proceduralna

W literaturze obsługi reklamacji sprawiedliwość jest rozkładana na wymiary: dystrybutywny (wynik), proceduralny (proces) i interakcyjny (traktowanie) (Blodgett i in., 1997; Tax i in., 1998). W badaniu Blodgetta, Hilla i Taxa (Blodgett i in., 1997) interakcyjny wymiar okazał się najsilniejszym predyktorem zachowań post-skargowych (repatronage, WOM) – choć wynik ten dotyczy sektora detalicznego i nie musi bezpośrednio przekładać się na inne branże. W praktyce rubryka ta obejmuje m.in. brak uznania sytuacji, protekcjonalność, dysonans ton–stawka, „skryptową grzeczność" w realnym kryzysie. W badaniu Blodgetta i in. klient, który doświadczał sprawiedliwego traktowania w interakcji (szacunek, uznanie problemu, empatia), oceniał całe doświadczenie obsługowe wyżej, nawet jeśli wynik merytoryczny był niesatysfakcjonujący (Blodgett i in., 1997). W kontekście konwersacyjnych systemów AI jest to szczególnie istotne: bot, który stosuje formułki empatyczne bez kontekstowego dopasowania (np. „Rozumiem, że to frustrujące" po raz trzeci w tej samej rozmowie), może naruszyć sprawiedliwość interakcyjną przez protekcjonalność.

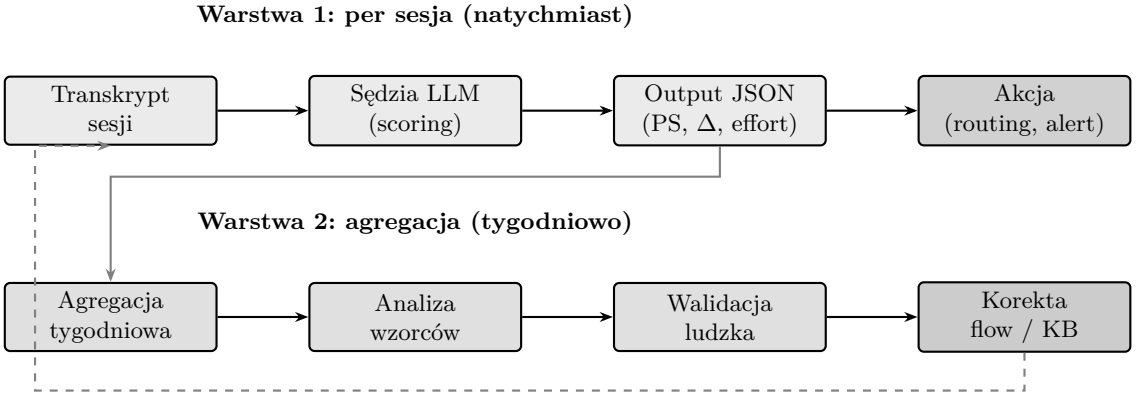
5.3.5. Dynamika: pętle, eskalacje, nieudane próby naprawy (failure recovery)

Czy rozmowa wpadła w pętlę (**No-Progress Loop**), w której klient sygnalizuje brak postępu, a system powtarza wariant tej samej odpowiedzi? Czy próba naprawy pogorszyła sytuację? Mechanizm eskalacji po nieudanej naprawie jest dobrze udokumentowany w badaniach service recovery (Smith i Bolton, 1998; Joireman i in., 2013; Bougie i in., 2003). W kontekście konwersacyjnych systemów AI pętle mają dodatkowy wymiar: w interakcji z człowiekiem agent może przynajmniej rozpoznać, że powtarza się, i zmienić podejście. Bot bez mechanizmu detekcji pętli może generować identyczne lub niemal identyczne odpowiedzi w nieskończoność, co tworzy sytuację, w której każda kolejna tura obniża walencję i zwiększa prawdopodobieństwo twardej eskalacji (żądanie zwrotu, groźba prawna, rezygnacja). W praktycznej implementacji detekcja pętli nie powinna opierać się wyłącznie na ocenie LLM. Model operujący na skompresowanym transkrypcie może przeoczyć pętlę rozgrywającą się w środku rozmowy, lub – odwrotnie – błędnie rozpoznać naturalne doprecyzowania jako powtórzenia. Bardziej niezawodnym podejściem jest **deterministyczny pre-pass algorytmiczny**, który generuje kandydatów na pętlę: dla każdej pary tur użytkownika w oknie czterech tur obliczamy podobieństwo leksykalne (np. Jaccard similarity na tokenach); para przekraczająca próg 0,5 jest traktowana jako *kandydat na powtórzenie intencji*. Kandydat staje się No-Progress Loop dopiero wtedy, gdy odpowiedź systemu między powtórzeniami nie dostarcza nowej, użytecznej informacji ani postępu zadaniowego (kanoniczna definicja `loop_count` – zob. tab. 4). Sama zbieżność leksykalna nie jest wystarczająca, ponieważ użytkownik może powtórzyć cel

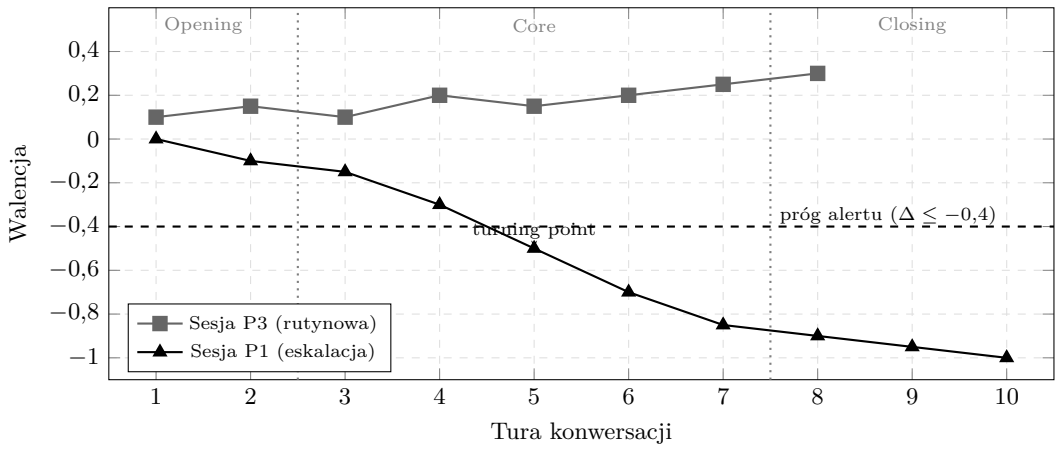
w sytuacji, gdy system między powtórzeniami dostarczył nową informację lub realnie przybliżył sprawę do rozwiązania. LLM służy jako weryfikator kontekstowy dla kandydatów z pre-passu oraz jako backup dla subtelnych przypadków – parafraz semantycznie tożsamych, których samo podobieństwo tokenów nie wychwyta.

5.3.6. Punkt zwrotny (turning point)

Sędzia nie tylko ocenia „wynik rozmowy", ale wskazuje **moment zmiany trajektorii**: gdzie nastąpiła eskalacja lub deeskalacja oraz co było bezpośrednim wyzwalaczem (np. brak odpowiedzi na konkretne pytanie, sprzeczność, obietnica bez pokrycia). To jest w praktyce najcenniejsze wyjście dla zespołów produktu i operacji: turning point pozwala przejść od oceny do poprawy procesu. Identyfikacja punktów zwrotnych jest zbieżna z metodologią *critical incident technique* (CIT) stosowaną w badaniach jakości usług (Flanagan, 1954; Lemon i Verhoef, 2016), gdzie kluczowe dla doświadczenia klienta są nie tyle „średnie" interakcje, co momenty przełomowe - zarówno pozytywne (exceeding expectations), jak i negatywne (service failure). W naszym podejściu automatyzujemy identyfikację tych momentów, co pozwala na analizę na skali niedostępnej dla tradycyjnego CIT.



Rysunek 4: Dwuwarstwowy model akcji. Warstwa 1 działa natychmiast per sesja (scoring → routing/alert). Warstwa 2 agreguje wyniki tygodniowo i wymaga walidacji ludzkiej przed wdrożeniem zmian w flow konwersacyjnym lub bazie wiedzy.



Rysunek 5: Przykładowe trajektorie walencji dla sesji P1 (eskalacja) i P3 (rutynowa) w podziale na fazy Opening / Core / Closing. Wartości walencji na osi Y odpowiadają skali $[-1, +1]$ i są liczone wyłącznie z tur użytkownika (tury asystenta są pomijane). Sesja P1 przekracza próg alertu ($\Delta \leq -0{,}4$) w fazie Core; turning point wskazuje moment zmiany trajektorii.

5.4. Struktura danych wyjściowych sędziego: podejście fazowe

Sędzia AI ma sens dopiero wtedy, gdy jego wynik jest **powtarzalny i agregowalny**. Dlatego wynik nie powinien być komentarzem opisowym, tylko **wypełnieniem schematu** (nawet jeśli w MVP będzie prosty). Przytoczone wcześniej podejście fazowe (**Opening / Core / Closing**) stanowi użyteczną podstawę MVP, ponieważ:

- jest stabilne między domenami i językami,
- pozwala policzyć trajektorię (np. delta opening→closing),
- daje szybkie rankingowanie rozmów (które poprawiły sytuację, a które ją pogorszyły).

Podejście fazowe jest również spójne z literaturą analizy dyskursu, gdzie konwersacje obsługowe mają rozpoznawalną strukturę fazową: otwarcie (identyfikacja problemu), rdzeń (próba rozwiązania) i zamknięcie (potwierdzenie/wyjście) (Drew i Heritage, 1992). Podział ten jest na tyle uniwersalny, że sprawdza się zarówno w rozmowach H2H, jak i Bot2H, co czyni go dobrą podstawą dla porównań międzykanałowych. Jednocześnie trzeba je „dociążyć" dwoma elementami, żeby nie było zbyt toporne:

1. **turning point** (punkt zwrotny), bo to on tłumaczy *dlaczego* delta wyszła taka, a nie inna,
2. **co najmniej jeden wskaźnik tarcia w core** (np. pętla / brak postępu / eskalacja po naprawie), bo w przeciwnym razie core jest czarną skrzynką.

To łączy prostotę MVP z minimalną zdolnością diagnostyczną: delta mówi „czy", turning point mówi „gdzie i dlaczego".

5.5. Biasy sędziego i konieczność kalibracji

Jeśli traktujemy model jako instrument pomiaru, musimy przyjąć konsekwencję: instrument ma błędy systematyczne. W literaturze LLM-as-a-Judge opisuje się m.in. bias pozycji (preferencja odpowiedzi podanej jako pierwsza), bias rozwlekłości oraz tendencję do auto-faworyzowania rozwiązań podobnych do własnego stylu (Zheng i in., 2023; Liu i in., 2023). Zheng i in. (2023) wykazali, że GPT-4 jako sędzia osiąga ponad 80% zgodności z ludzkimi ocenami w benchmarkach konwersacyjnych, ale tylko pod warunkiem zastosowania kontroli pozycyjnej (losowanie kolejności prezentowania odpowiedzi) i jasno zdefiniowanych kryteriów. W naszym kontekście dochodzi dodatkowe obciążenie specyficzne dla domeny obsługowej: LLM mogą faworyzować odpowiedzi bota, które są „grzeczne" i „empatyczne" na poziomie powierzchniowym, nawet jeśli nie rozwiązują problemu — jest to odmiana obciążenia rozwlekłości (*verbosity bias*). Osobnym, choć pokrewnym zjawiskiem jest *sycophancy bias*, tj. tendencja modelu do dostosowywania odpowiedzi do oczekiwań użytkownika zamiast zachowania niezależnej oceny (Sharma i in., 2023). W praktyce wdrożeniowej redukujemy to przez zestaw prostych zasad:

- **stabilne rubryki** (te same definicje w czasie) i wersjonowanie promptów,
- **wymóg dowodu** (spany + uzasadnienie),
- **izolacja kontekstu między warstwami pipeline'u** - w architekturze wielowarstwowej (klasyfikacja → ocena outcome → scoring trajektorii) każda warstwa powinna operować we własnym kontekście konwersacyjnym; wyniki poprzednich warstw przekazywane są jako dane strukturalne, nie jako wcześniejsze wypowiedzi modelu. W przeciwnym razie model może „bronić" błędnej klasyfikacji z warstwy 1, traktując ją jako własne wcześniejsze twierdzenie (*anchoring bias*),
- **kontrole symetrii** (np. losowanie kolejności fragmentów, gdy porównujemy warianty),
- **audyt próby przez człowieka** (do kalibracji i wykrywania driftu),
- **monitoring driftu**: język użytkowników, polityki i produkty się zmieniają; sędzia musi być rekalibrowany.

Dodatkowo, w kontekście wielojęzycznym i wielokulturowym, konieczne jest uwzględnienie faktu, że normy komunikacyjne (np. dopuszczalność bezpośredniej krytyki, oczekiwana formalność, interpretacja milczenia) różnią się między kulturami, co wpływa na kalibrację walencji (Hofstede i in., 2010). W MVP ograniczamy ten problem do jednego języka; rozszerzenie na kolejne wymaga osobnej kalibracji rubryk. Powyższe zasady stanowią fundament inżynierii jakości systemu: sędzia działa tylko tak dobrze, jak dobrze zaprojektowane są rubryki i nadzór nad nimi.

6. KPI: Sterowanie produktem i operacjami

Jeżeli rozdział 5 ustanawia instrument, to rozdział 6 odpowiada na pytanie: **jakie wskaźniki z tego instrumentu są zasadne dla osób zarządzających**. Kluczowym kryterium doboru KPI jest ich przydatność decyzyjna. Oznacza to spełnienie w tym przypadku trzech warunków:

1. są policzalne na dużej skali,
2. są stabilne między kanałami (H2H/Bot2H),
3. mają jasne przełożenie na decyzje: routing, zmiana flow, poprawa bazy wiedzy, eskalacja do człowieka.

W oparciu o nasze doświadczenia wdrożeniowe (8 374 wiadomości, 1 146 przefiltrowanych sesji) traktujemy klasyczny sentyment jako sygnał przydatny, ale **potencjalnie niewystarczający**. Nie dlatego, że „jest zły", tylko dlatego, że walencja rzadko wskazuje mechanizm problemu: czy zawiódł fakt, procedura, czy komunikacja. Standardowe automatyczne metryki NLG mogą wykazywać ograniczoną zgodność z ocenami ludzi, szczególnie w zadaniach wymagających oceny wielu wymiarów jakości jednocześnie (Liu i in., 2023). Sentyment jest jeszcze węższym sygnałem, ponieważ opisuje głównie walencję, a nie mechanizm problemu. W obsłudze klienta to rozróżnienie jest fundamentalne — rozmowa może mieć pozytywny sentyment na zamknięciu (klient uprzejmie się żegna), ale być porażką operacyjną (problem nierozwiązany, klient odchodzi „po cichu"). Poniżej prezentuję KPI pogrupowane w trzy koszyki: **wynik**, **tarcie/ryzyko**, **zaufanie**.

6.1. Wskaźniki wyniku: spełnienie intencji i kompletność rozwiązania

6.1.1. Intent Fulfillment Rate (IFR)

Odsetek rozmów, w których intencja została domknięta (true / partial / false). IFR pełni funkcję zbliżoną do operacyjnego wskaźnika *First Contact Resolution* (FCR), choć jest wnioskowany wyłącznie z transkryptu i nie stanowi potwierdzenia faktycznego rozwiązania sprawy w systemach backendowych. Ważne jest rozróżnienie między „true" a „partial": częściowe domknięcie intencji (np. status zamówienia podany, ale pytanie o rekompensatę zignorowane) może zwiększać prawdopodobieństwo ponownego kontaktu (*recontact*) z wyższą frustracją startową, ponieważ użytkownik musi ponownie inwestować wysiłek w coś, co „powinno było być załatwione za pierwszym razem". Kategoria „partial" jest jednak niejednorodna operacyjnie i wymaga dalszego rozróżnienia. **Helpful partial** - bot zrobił realny postęp, część problemu faktycznie domknięta - to zupełnie inny sygnał niż **deflected** - bot przekierował bez pomocy (odesłał do infolinii, powtórzył link FAQ) - lub **apparently_misinformed** - sędzia na podstawie transkryptu ocenia, że bot mógł podać błędne informacje przyjęte przez klienta jako prawdziwe. Trzeci przypadek jest najpoważniejszy: klient odchodzi z przekonaniem, że problem jest rozwiązany, a discovery następuje później i generuje gwałtowną eskalację trudną do odwrócenia. (Ostateczne potwierdzenie wymaga weryfikacji z bazą wiedzy operatora — pole należy do warstwy rozszerzonej.)

6.1.2. Recontact / Reopen Rate (w zależności od kanału)

Jeśli mamy identyfikator użytkownika lub sprawy, ten wskaźnik mierzy, czy rozmowa faktycznie rozwiązała problem, czy tylko go odłożyła. Jest to wskaźnik *outcome*, nie *process* - nie zależy od jakości pomiaru sędziego, tylko od zachowania użytkownika po interakcji. Wysoki recontact rate przy wysokim IFR sugeruje, że sędzia źle ocenia fulfillment (problem kalibracji). Niski recontact rate przy niskim IFR sugeruje, że klienci odchodzą zamiast wracać (silent withdrawal) - co jest gorszym scenariuszem biznesowym.

6.1.3. Time-to-Resolution / Turns-to-Resolution

Wskaźniki „czas" i „liczba tur" są niedoskonałe (czasem sprawa jest trudna), ale jako trend w obrębie tej samej kategorii intencji świetnie pokazują degradację procesu. Nagły wzrost średniej liczby tur dla intent=billing_issue może wskazywać na zmianę polityki, błąd w bazie wiedzy lub degradację modelu.

6.1.4. Proponowany rozszerzony Resolution Quality Score

Syntetyczny wynik rubryk agregujący wymiary jakości rozwiązania. W obecnym MVP wskaźnik ten opiera się wyłącznie na sygnałach dostępnych z transkryptu:

MVP: intent fulfillment (transcript-inferred), effort score, trajectory delta, loop detection, exit intent.

Warstwa rozszerzona (wymaga zewnętrznych źródeł prawdy): poprawność merytoryczna (factual correctness), zgodność z polityką (policy compliance), promise_supported (weryfikacja realizacji obietnicy), potwierdzone resolution (na podstawie danych backendowych).

Rozróżnienie jest istotne: obecny system analizuje przebieg rozmowy, a nie niezależnie potwierdzoną poprawność agenta. Pełny Resolution Quality Score, agregujący oba poziomy, wymaga integracji ze źródłami prawdy (baza wiedzy, systemy ticketowe) i stanowi cel kolejnej iteracji.

6.2. Wskaźniki tarcia i kosztu poznawczego

Poniższe wskaźniki przesuwają perspektywę z pytania o skuteczność rozwiązania na pytanie o koszt poznawczy ponoszony przez użytkownika.

6.2.1. Effort Score (proxy dla CES)

Skala 1–5, gdzie 5 oznacza wysokie tarcie: powtarzanie, dopytywanie, brak postępu. KPI nadaje się do dashboardów i do priorytetyzacji kategorii wymagających poprawy. Customer Effort Score (CES) jest w literaturze obsługowej konsekwentnie wymieniany jako silny predyktor lojalności - Dixon i in. (2013) argumentują, że redukcja wysiłku klienta ma większy wpływ na lojalność niż „zachwycanie" (delight). W naszym modelu CES jest estymowane z transkryptu (proxy), a nie z ankiety - co pozwala na pomiar każdej rozmowy, nie tylko tych, w których klient wypełnił feedback.

6.2.2. Loop Rate (częstość pętli)

Odsetek rozmów (albo tur), w których wykryto pętlę: powtarzalność odpowiedzi mimo sygnałów braku postępu. Pętle są operacyjnie ważne, bo generują eskalację i porzucenia nawet wtedy, gdy język klienta pozostaje względnie „poprawny". To czyni je „cichymi zabójcami" satysfakcji - tradycyjny sentiment analysis ich nie łapie, bo klient może uprzejmie powtarzać prośbę, jednocześnie budując frustrację. W implementacji warto pamiętać, że detekcja pętli oparta wyłącznie na LLM jest podatna na dwa błędy: (1) przeoczenie pętli rozgrywającej się w środku długiej sesji - jeśli pipeline kompresuje transkrypt, środkowe tury mogą zostać pominięte; (2) fałszywy alarm przy naturalnym doprecyzowaniu pytania. Bardziej niezawodne jest podejście hybrydowe z deterministycznym pre-passem algorytmicznym, opisane szczegółowo w sekcji 5.3(5).

6.2.3. Failure-Recovery Breakdown Rate

Odsetek przypadków, w których próba naprawy pogorszyła trajektorię (np. po przeprosinach i „uspokajaniu" klient przechodzi do silniejszej krytyki). To KPI stoi wprost na literaturze service recovery: jakość naprawy bywała oceniana surowiej niż sam pierwotny błąd (Smith i Bolton, 1998; Joireman i in., 2013). Operacyjnie ten wskaźnik jest szczególnie cenny, bo wskazuje na problemy z **playbookami naprawy** - jeśli Failure-Recovery Breakdown Rate rośnie dla konkretnej kategorii intencji, to sygnał, że strategia recovery (ton, treść, kolejność kroków) wymaga korekty.

6.3. Wskaźniki trajektorii relacyjnej

Jeśli mamy fazy **Opening/Core/Closing**, dostajemy prostą, ale mocną geometrię trajektorii.

6.3.1. Opening→Closing Delta (Δ)

Różnica między oceną fazy początkowej i końcowej (np. w skali numerycznej albo ordinalnej). To KPI „kierunku zmiany".

- $\Delta > 0$: interakcja poprawiła sytuację,
- $\Delta < 0$: interakcja pogorszyła sytuację.

Delta jest wskaźnikiem **porównawczym** (a nie kontrfaktycznym): odpowiada na pytanie „czy po rozmowie z nami jest lepiej czy gorzej niż przed?". W odróżnieniu od sentymentu zamknięcia (który może być negatywny, bo klient przyszedł z problemem, ale rozmowa poprawiła sytuację), delta mierzy zmianę. W sesjach z jedną turą użytkownika opening = closing, więc Δ wynosi trywialnie 0 – takie sesje otrzymują $\Delta :=$ NA i są pomijane w analizach trajektorii.

6.3.2. Core Minimum (najgorszy punkt rozmowy)

Minimalny wynik w trakcie rozmowy (najbardziej ryzykowny moment). To KPI działa jak wykrywacz „prawie-katastrof": rozmowa może skończyć się poprawnie, ale jeśli miała bardzo niski dołek, to jest kandydat do analizy turning pointów. Core Minimum jest ważny, bo delta może kłamać: rozmowa, która zaczęła się neutralnie (0.0), spadła do -0.8 w core i wróciła do -0.1 na zamknięciu, ma Δ = -0.1 (niewielkie pogorszenie), ale miała „prawie-katastrofę" w środku. Bez core_min ta informacja jest niewidoczna. Kluczowa uwaga implementacyjna: Core Minimum powinno być liczone **wyłącznie z tur użytkownika**, nie z wszystkich wiadomości w sesji. Tury asystenta są zazwyczaj neutralne (walencja ≈ 0.0) - włączenie ich do obliczenia rozmywa sygnał i zaniża rzeczywisty dołek emocjonalny klienta. To samo dotyczy Opening i Closing Delta: obie wartości powinny średnić wyłącznie turny użytkownika, nie wszystkich uczestników rozmowy.

6.3.3. Turning Point Rate i typologia triggerów

Odsetek rozmów z wykrytym punktem zwrotnym oraz najczęstsze wyzwalacze (sprzeczność, brak konkretu, obietnica bez pokrycia, pętla). To KPI jest bezpośrednio „produkto-twórcze": mówi, co poprawiać w flow i bazie wiedzy. Agregacja typologii triggerów (np. „35% turning pointów wynika z obietnic bez pokrycia w kategorii refund_request") daje zespołom produktowym konkretny backlog poprawek, priorytetyzowany częstością występowania i wagą skutków. To przejście od „mamy 15% niezadowolonych rozmów" do „mamy 15% niezadowolonych rozmów, z czego 35% wynika z konkretnego problemu X, który można naprawić zmianą Y".

6.4. Wskaźniki zaufania i zgodności

W obsłudze (zwłaszcza reklamacyjnej i płatniczej) ryzyko nie wynika wyłącznie z emocji, tylko z tego, czy system mówi rzeczy prawdziwe i wykonalne.

6.4.1. Hallucination / Unsupported Claim Rate

[Warstwa rozszerzona – wymaga zewnętrznych źródeł prawdy.] Odsetek odpowiedzi, które składają obietnice lub stwierdzenia bez pokrycia w politykach lub danych. W service recovery to jest paliwo eskalacji: druga porażka boli bardziej (Smith i Bolton, 1998; Joireman i in., 2013). W kontekście konwersacyjnych systemów AI halucynacje mają szczególną wagę, ponieważ użytkownik często nie jest w stanie odróżnić halucynacji od poprawnej odpowiedzi w momencie interakcji – odkrycie następuje później (np. gdy obiecany zwrot nie nadchodzi), co generuje gwałtowny spadek zaufania i często twarde eskalacje (chargeback, skarga do regulatora). W obecnym MVP wskaźnik ten jest pominięty (por. sekcja 7.2); jego operacjonalizacja wymaga dostępu sędziego do bazy wiedzy lub polityk, co wykracza poza architekturę opartą wyłącznie na transkrypcie.

6.4.2. Evidence Adherence Score

[Warstwa rozszerzona – wymaga zewnętrznych źródeł prawdy.] Czy odpowiedź opiera się na źródle (KB/polityka), a nie na „gładkiej narracji". Ten KPI daje się audytować i jest szczególnie ważny w obszarach wysokiej stawki. Stanowi bezpośrednie przełożenie zasady *evidence-only* z poziomu sędziego na poziom ocenianego systemu: tak jak sędzia musi uzasadnić swoją ocenę, tak bot/agent powinien uzasadnić swoją odpowiedź. Analogicznie do Hallucination Rate, pełna operacjonalizacja wymaga integracji ze źródłami prawdy.

6.4.3. Justice Scores (procedural / interactional / distributive)

Jeśli w MVP nie chcemy pełnych trzech wymiarów, zaczynamy od dwóch: proceduralnej i interakcyjnej. Literatura pokazuje, że interakcyjna sprawiedliwość ma silny wpływ na zachowania po skardze (WOM, repatronage) (Blodgett i in., 1997; Tax i in., 1998). W praktyce proceduralny justice score odpowiada na pytanie: „Czy użytkownik wie, co się dzieje i dlaczego?" (transparentność procesu), a interakcyjny: „Czy użytkownik jest traktowany z szacunkiem i uznaniem?". Dystrybutywny (czy wynik jest uczciwy) może być dodany w kolejnej iteracji.

6.4.4. Retaliation Risk Flags (proxy)

Nie chodzi o „diagnozę gniewu", tylko o wykrycie wzorców językowych i intencji odwetu (zapowiedź skargi, publikacji, odejścia). Badania marketingowe pokazują, że silna złość w usługach bywa powiązana z zachowaniami odwetowymi, a nie tylko z rezygnacją (Bougie i in., 2003). W badaniu Bougie i in. (2003) złość (w odróżnieniu od rozczarowania) aktywowała motywację konfrontacyjną – klienci w stanie złości byli bardziej skłonni do negatywnego WOM i składania skarg niż do pasywnego wycofania. Choć badanie nie mierzyło bezpośrednio skarg do regulatorów ani aktywności w mediach społecznościowych, mechanizm konfrontacyjny sugeruje, że ryzyko może rozciągać się na te kanały. Dlatego detekcja sygnałów odwetowych (nawet jeśli niedoskonała) ma bezpośrednią wartość w zarządzaniu ryzykiem reputacyjnym.

6.5. Progi alarmowe i mechanizmy sterowania operacyjnego

KPI bez progów są opisem. KPI z progami są mechanizmem zarządzania ryzykiem. Praktycznie działa prosty układ:

- **zielony / żółty / czerwony**,
- progi **absolutne** (np. halucynacje w płatnościach) i **względne** (np. 1,5× wzrost Loop Rate vs baseline),
- reakcje w trzech klasach:
  1. **routing** (bot → człowiek → senior service support),
  2. **tryb odpowiedzi** (source-required, no-promises, action-first),
  3. **kill-switch** (wyłączenie automatyzacji dla ścieżek o wysokim koszcie błędu).

Uwaga metodologiczna: w danych obsługowych ryzyko bywa nieliniowe. Niewielki wzrost wskaźników tarcia może wywołać skokowy wzrost eskalacji, zwłaszcza gdy uruchamia się „zawód po naprawie" (failure recovery breakdown). Jest to zbieżne z modelem kumulacji błędów predykcyjnych (Clark, 2016): po przekroczeniu indywidualnego progu tolerancji następuje gwałtowna zmiana strategii użytkownika. Dlatego progi nie powinny być statyczne. W iteracji post-MVP rekomendujemy dynamiczne progi oparte na percentylach baseline'u per kategoria intencji: jeśli Loop Rate dla billing_issue historycznie wynosi 8%, próg żółty na 12% i czerwony na 18% ma sens; ale te same wartości bezwzględne mogą być normalne lub alarmujące w innej kategorii.

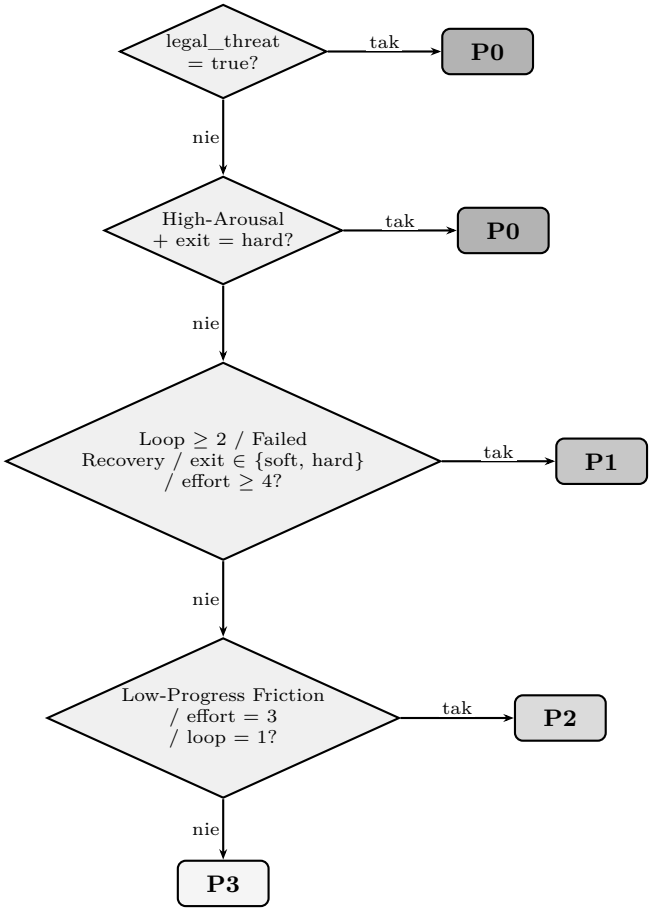
6.6. Minimalny zestaw wskaźników MVP

W kontekście wdrożenia MVP minimalny zestaw wskaźników, spójny z rubrykami opisanymi w rozdziale 5, obejmuje następujące elementy:

1. **Δ Opening→Closing** (trajektoria - liczona z tur użytkownika)
2. **Intent fulfillment** (true/partial/false) + **fulfillment quality** dla partial (helpful_partial / deflected / apparently_misinformed)
3. **Effort score** (1–5)
4. **Loop detected + loop count** (z algorytmicznym pre-passem jako podstawą)
5. **Turning point (index + trigger)**
6. **Exit intent level** (null / soft / hard) – gradient zamiast boolean; osobne pole `legal_threat: boolean` (groźba prawna może współwystąpić z dowolnym poziomem exit intent)
7. **Promise detection flag** (detekcja jawnej obietnicy w transkrypcie; weryfikacja spełnialności wymaga warstwy rozszerzonej)

Powyższy zestaw spełnia następujące kryteria:

- jest prosty,
- daje ranking „najgorszych rozmów",



Rysunek 6: Drzewo decyzyjne przypisania Priority Score (P0–P3). Wszystkie progi eskalacji są zawarte w jednej tabeli decyzyjnej (zob. sekcja 4.2).

- tłumaczy przyczynę (turning point),
- pozwala odróżnić problem merytoryczny od proceduralnego i relacyjnego.

Celowo pomijamy w MVP pełne justice scores i hallucination rate jako osobne KPI – justice gap manifestuje się pośrednio w effort i delta, natomiast hallucination rate i evidence adherence score należą do warstwy rozszerzonej, wymagającej integracji ze źródłami prawdy (baza wiedzy, polityki) – w architekturze opartej wyłącznie na transkrypcie ich operacjonalizacja nie jest możliwa. Pole `promise_detected` sygnalizuje jedynie obecność jawnej obietnicy w wypowiedzi bota; ocena, czy obietnica była spełnialna (`promise_supported`), wymaga dostępu do zewnętrznego źródła prawdy i należy do warstwy rozszerzonej. W kolejnej iteracji mogą stać się osobnymi wskaźnikami. W kolejnym kroku możemy to przetłumaczyć na **finalny schemat zbierania metryk**: jednostki analizy, okna, definicje progów, format outputu i minimalny model danych.

7. Plan walidacji empirycznej

Framework opisany w niniejszym artykule działa w środowisku produkcyjnym i zbiera dane w trybie ciągłym. Poniżej przedstawiamy zaplanowany protokół walidacji, którego wyniki zostaną opublikowane jako aktualizacja niniejszego preprintu oraz jako osobny artykuł empiryczny. **Cel 1: Inter-rater agreement dla kategorii taksonomii.** [IN PROGRESS] Zaplanowane są dwa komplementarne zbiory anotacyjne: (a) **próba reprezentatywna** – 200+ sesji wylosowanych z korpusu, stratyfikowanych po domenie i długości sesji, służąca do estymacji częstości populacyjnych i ogólnego poziomu zgodności; (b) **próba wzbogacona** (*enriched sample*) – celowo dobrane sesje zawierające kandydackie przypadki rzadkich zdarzeń (P0, Exit Intent, No-Progress Loop, Justice Gap), służąca do mierzenia recall i zgodności anotatorów dla klas o niskiej częstości naturalnej. Precision i PR-AUC będą estymowane na próbie reprezentatywnej lub na próbie wzbogaconej po zastosowaniu wag odtwarzających naturalną częstość klas (surowe precision z próby wzbogaconej jest artefaktem sztucznie zmienionej proporcji). Przy loop rate $\approx 0{,}4\%$ losowa próba 200 sesji może nie zawierać ani jednego przypadku pętli – stąd konieczność wzbogacenia. Z próby wzbogaconej nie wolno bezpośrednio wyliczać częstości populacyjnych.

Trzy osoby z doświadczeniem w analizie obsługi klienta będą niezależnie anotować każdą sesję według rubryk taksonomii (klasyfikacja kategorii, effort score, trajektoria, turning point, intent fulfillment). Codebook z definicjami operacyjnymi, kryteriami koniecznymi i wykluczającymi oraz przykładami pozytywnymi i negatywnymi zostanie opracowany przed anotacją i przetestowany na próbie pilotażowej (15 sesji). Metryka: Krippendorff's alpha per kategoria; dwupoziomowy próg akceptacji: $\alpha \geq 0{,}67$ dla fazy eksploracyjnej (wystarczający do dalszych analiz i iteracji rubryk), $\alpha \geq 0{,}80$ dla fazy operacyjnej (wymagany przed użyciem wyników do routingu decyzji). Dla kategorii krytycznych (P0, P1) dodatkowo wymagane minimum: recall $\geq 0{,}80$ i precision $\geq 0{,}70$ – asymetria odzwierciedla wyższy koszt fałszywego negatywu (pominięcie sesji wysokiego ryzyka) niż fałszywego alarmu. Docelowo próg decyzyjny zostanie wybrany na podstawie jawnie określonej funkcji kosztu fałszywie ujemnych i fałszywie dodatnich decyzji, skalibrowanej na danych z walidacji (Cel 2). *Status: trwa rekrutacja anotatorów i przygotowanie codebooka.* **Cel 2: Zgodność LLM vs. człowiek (concurrent validity).** [IN PROGRESS] Ten sam zbiór sesji z anotacjami ludzkimi zostanie scorowany przez sędziego LLM według identycznych rubryk. Porównanie per rubryka: precision, recall, F1 dla kategorii binarnych/multilabel; weighted Cohen's kappa (wagi liniowe) dla Priority Score (skala porządkowa P0–P3) oraz effort score (skala porządkowa 1–5, faktycznie wykorzystująca trzy wartości); ICC pomocniczo dla trajectory delta (jedyna skala o charakterze quasi-ciągłym). Macierz pomyłek per kategoria pozwoli zdiagnozować nie tylko ile sędzia się myli, ale jak się myli. Benchmark: F1 ≥ 0.70 per kategoria, weighted $\kappa \geq 0.70$ dla PS i effort score, ICC ≥ 0.75 dla trajectory delta. *Uwaga metodologiczna:* walidacja kategorii Justice Gap (obecnej w taksonomii, ale nie posiadającej dedykowanego pola w obecnym schemacie JSON) zostanie przeprowadzona na rozszerzonej wersji schematu zawierającej dedykowane pole `justice_gap`; bez tego rozszerzenia porównanie LLM–człowiek dla tej kategorii nie jest wykonalne. *Status: zaplanowane po zakończeniu anotacji ludzkiej (Cel 1).* **Cel 3: Test-retest reliability.** 50 sesji z próby zostanie scorowanych przez sędziego LLM pięciokrotnie w dwóch wariantach: (a) z temperaturą 0,0 (konfiguracja produkcyjna) – weryfikacja stabilności wyników i ewentualnej niedeterministyczności backendu API; (b) z temperaturą > 0 – stress test odporności rubryk na stochastyczność modelu. Mierzony będzie weighted κ dla Priority Score i effort score (skale porządkowe) oraz ICC dla trajectory delta. Próg akceptacji: weighted $\kappa \geq 0{,}80$, ICC $\geq 0{,}80$. Źródła wariancji (prompt vs. ambiguity transkryptu vs. stochastyczność modelu) będą diagnozowane w przypadku rubryk poniżej progu. **Cel 4: Walidacja predykcyjna.** System produkcyjny zbiera dane o rezultatach zewnętrznych (ponowny kontakt, churn, CSAT) równolegle z ocenami sędziego. Zaplanowana jest analiza korelacji w oknach zdefiniowanych osobno

per metrykę rezultatu (ponowny kontakt: 7 dni, CSAT: 14 dni, churn: 30–90 dni): regresja logistyczna z AUC-ROC na danych z niezależną etykietą rezultatu, porównanie z baseline'ami (sam sentyment, proste metryki procesowe). Hipoteza: wyniki frameworku przewidują negatywny rezultat zewnętrzny lepiej niż sam sentyment. Wstępny test wewnętrzny (tabela 8) pokazuje większą separację etykiet dla frameworku, jednak zawiera target leakage (exit_intent wchodzi do obu stron) – wynik traktujemy jako przesłankę do dalszych badań, nie jako dowód walidacyjny. **Cel 5: Test ekwiwalencji tłumaczeniowej (*translation-equivalence stress test*).** 50 sesji z korpusu polskiego zostanie przetłumaczonych na język angielski (tłumaczenie profesjonalne z zachowaniem pragmatyki). Sędzia LLM ocenia obie wersje; porównanie wyników pozwoli zdiagnozować, gdzie kalibracja się rozjeżdża między wersjami językowymi. *Uwaga:* ten test sprawdza inwariancję wyników względem tłumaczenia, nie trafność na języku polskim. Podstawową walidacją polskojęzyczną jest zgodność LLM–człowiek na oryginalnych transkryptach (Cel 2); Cel 5 nie jest jej substytutem.

Zabezpieczenia proceduralne: (1) *Zaślepienie anotatorów:* anotatorzy nie widzą ocen sędziego LLM, Priority Score, confidence ani zewnętrznego outcome'u – w przeciwnym razie ocena ludzka mogłaby nieświadomie zbliżyć się do wyniku modelu. (2) *Zamrożenie instrumentu:* przed oceną końcową (Cel 2) zostaną zamrożone: codebook, prompt, rubryki, reguły Priority Score i progi decyzyjne – próba używana do iteracji rubryk nie może jednocześnie stanowić zbioru testowego. (3) *Plan liczebności:* 200+ sesji jest rozsądną próbą dla pilotażu anotacji (Cel 1–2), ale niekoniecznie dla walidacji outcome'ów (Cel 4), gdzie rzadkość zdarzeń P0 i churn wymaga większych zbiorów. Wielkość próby dla walidacji zewnętrznych outcome'ów zostanie ustalona na podstawie oczekiwanej częstości zdarzeń i wymaganej precyzji przedziałów ufności.

Doprecyzowania metodologiczne: Docelowa minimalna liczba pozytywnych przypadków per rzadka klasa w próbie wzbogaconej: ≥ 30 (minimalna próba pilotażowa umożliwiająca wstępną analizę błędów; precyzyjne estymaty F1 i recall będą wymagały większych zbiorów). Procedura rozstrzygania sporów anotatorów: mediacja z trzecim anotatorem; w przypadku trwałej rozbieżności sesja otrzymuje etykietę `ambiguous` i pozostaje w zbiorze. Wyniki raportowane będą dwukrotnie: na pełnym zbiorze (z przypadkami `ambiguous`) oraz na podzbiorze z pełną zgodnością – co pozwoli ocenić wpływ niejednoznacznych przypadków na metryki i zdiagnozować granice taksonomii. Odsetek sporów będzie raportowany per kategoria. **Metryki audytowalności:** Turning point będzie oceniany w dwóch trybach: exact match (zgodność co do numeru tury) oraz z tolerancją ± 1 tura; trigger type – macro-F1; evidence spans – token-level F1 lub IoU (overlap między spanem wskazanym przez sędziego a spanem referencyjnym) plus ocena ludzka, czy wskazany fragment rzeczywiście uzasadnia przypisaną etykietę. W walidacji predykcyjnej (Cel 4): oprócz ROC-AUC raportowany będzie PR-AUC (istotny przy nierównowadze klas, 21% pozytywnych), ocena kalibracji (calibration plot + Brier score) oraz holdout temporalny lub domenowy (zbiór do strojenia rubryk musi być rozdzielony od zamrożonego zbioru testowego). Okno outcome zdefiniowane osobno per metrykę: recontact 7 dni, CSAT 14 dni, churn 30–90 dni.

7.1. Wstępne obserwacje z systemu produkcyjnego

System działa w środowisku produkcyjnym i przetworzył dotychczas **1 146 sesji** (przefiltrowane: min. 2 wiadomości). Poniższe obserwacje mają charakter wstępny i nie zastępują formalnej walidacji opisanej powyżej, jednak stanowią pierwszy sygnał o zachowaniu frameworku na danych produkcyjnych.

PS	n	false	partial	true	Negatywny sentyment
P0	36	25%	61%	14%	100%
P1	119	15%	85%	**0%**	**100%**
P2	55	24%	76%	0%	93%
P3	936	18%	48%	**33%**	25%

Tabela 5: Wewnętrzna spójność taksonomii – spełnienie intencji i sentyment zamknięcia per Priority Score ($n = 1\,146$ sesji). Dane pochodzą z ocen sędziego LLM, nie z niezależnej anotacji.

Żadna ze 119 sesji P1 nie zakończyła się pełną realizacją intencji – co jest spójne z założeniami taksonomii, choć wymaga potwierdzenia na niezależnie anotowanej próbie.

Diagnostyczny test separacji etykiet (AUC-ROC) został przeniesiony do Aneksu C, ponieważ zawiera target leakage i nie stanowi walidacji predykcyjnej.

Obserwacje dot. jakości danych:

- **resolution_status jest redundantne z intent_fulfillment**: mapping 1:1 (resolved=true, partially_resolved=partial, unresolved=false) - jedno z pól można usunąć z pipeline'u.
- **failed_recovery = false dla 100% sesji** – brak wariancji. Oznacza to, że pole to w obecnej wersji nie wnosi żadnej informacji diagnostycznej. Prawdopodobne przyczyny: zbyt restrykcyjna definicja w prompcie sędziego, błąd w pipeline, lub faktyczny brak zdarzeń tego typu w korpusie. Do wyjaśnienia w ramach walidacji (Cel 1–2).
- **promise_detected = 0% sesji** – analogicznie, brak wariancji. Pole zostało zdefiniowane jako detekcja jawnej obietnicy w transkrypcie (warstwa bazowa); ocena spełnialności obietnicy (`promise_supported`) wymaga zewnętrznych źródeł prawdy (SLA, stany magazynowe) i należy do warstwy rozszerzonej. Zerowa wariancja wymaga audytu: albo sędzia stosuje zbyt restrykcyjną definicję obietnicy, albo boty w korpusie faktycznie nie składają jawnych deklaracji.
- **effort_score efektywnie trinarny**: skala wynosi 1–5, jednak wartości 4 i 5 nie wystąpiły w żadnej z 1 146 sesji (86,8% effort=3, 11,7% effort=1, 1,5% effort=2). Rozkład ten sugeruje, że sędzia LLM de facto stosuje trzy poziomy trudności - wymaga to kalibracji promptu lub redefinicji skali.
- **confidence sędziego**: średnia = 0,856, mediana = 0,850; 98,6% sesji $\geq 0{,}85$. Wartości te są deklaracjami modelu (*self-reported*) i nie zostały dotąd skalibrowane względem rzeczywistej poprawności ocen. Model może być jednakowo pewny zarówno trafnych, jak i błędnych werdyktów – dopóki confidence nie zostanie porównane z anotacją ludzką (Cel 2), nie należy go interpretować jako prawdopodobieństwa poprawności.

Zarządzanie danymi. Transkrypty sesji przed przekazaniem do pipeline'u scoringowego podlegają pseudonimizacji: imiona, nazwiska, adresy e-mail, numery telefonów i numery zamówień są zastępowane tokenami typu `[CUSTOMER_NAME]`, `[ORDER_ID]`. Scoring odbywa się za pośrednictwem API modelu zewnętrznego (OpenAI GPT-4o); transkrypty są przesyłane zgodnie z polityką przetwarzania danych i retencji skonfigurowaną dla konta produkcyjnego i nie są wykorzystywane do treningu modelu dostawcy. W przypadku wdrożeń, w których Zero Data Retention (ZDR) zostało zagwarantowane kontraktowo i technicznie, fakt ten jest dokumentowany osobno. Wyniki scoringu (JSON) są przechowywane w bazie danych operatora z retencją 12 miesięcy, po czym podlegają automatycznemu usunięciu, chyba że obowiązek prawny wymaga dłuższego przechowywania. Dostęp do danych surowych jest ograniczony do zespołu analitycznego; dashboard prezentuje wyłącznie zagregowane metryki. W kontekście domeny medycznej (obecnej w korpusie) stosowane są dodatkowe środki ostrożności: korpus medyczny ograniczono do rozmów administracyjnych (rejestracja, odwoływanie wizyt) i nie obejmuje opisów diagnoz, objawów ani leczenia. Dane są pseudonimizowane, lecz nadal traktowane jako dane osobowe; ich ewentualna kwalifikacja jako danych dotyczących zdrowia w rozumieniu art. 9 GDPR wymaga oceny zależnej od kontekstu (sam fakt korzystania z usługi medycznej może w pewnych okolicznościach ujawniać informacje dotyczące zdrowia). Aktualne wytyczne dotyczące granic między pseudonimizacją a anonimizacją zawiera dokument EDPB Guidelines 01/2025 on Pseudonymisation[1].

Uwaga etyczna. System de facto profiluje użytkowników pod kątem ryzyka relacyjnego na podstawie ich wypowiedzi tekstowych. Choć stosowana jest zasada Evidence-Only (ocena opiera się na obserwowalnych markerach, nie na spekulacji o stanach wewnętrznych), automatyczne scorowanie konwersacji rodzi pytania o prywatność i autonomię użytkownika. W kontekście europejskim istotne mogą być wymogi GDPR oraz Rozporządzenia (UE) 2024/1689 (AI Act). Zakres obowiązków przejrzystości, praw osób, których dane dotyczą, oraz nadzoru ludzkiego zależy od kontekstu wdrożenia i od tego, czy scoring prowadzi do decyzji wywołującej skutki prawne lub podobnie istotnie wpływającej na osobę (art. 22 GDPR nie stosuje się automatycznie do każdego wewnętrznego scoringu konwersacji). Przed wykorzystaniem Priority Score do zautomatyzowanych decyzji wpływających na użytkowników wymagana jest dedykowana ocena prawna. Rekomendujemy, by wdrożenia produkcyjne jawnie informowały użytkowników o scoringu.

8. Wnioski

Niniejszy artykuł przedstawia hipotezę, że dominujące sposoby mierzenia jakości interakcji w systemach conversational AI - sentyment per wiadomość, defleksja, czas odpowiedzi - nie uchwytują zjawisk, które mogą mieć istotne znaczenie dla wyniku relacyjnego: braku postępu w dialogu, jakości próby naprawy błędu czy cichego odejścia klienta bez skargi. Luka ta ma charakter operacyjny, nie tylko akademicki: prowadzi do tego, że systemy, które generują Double Deviation, mogą przez długi czas wyglądać dobrze w klasycznych raportach KPI.

[1] EDPB, *Guidelines 01/2025 on Pseudonymisation*, przyjęte 16 stycznia 2025; `https://www.edpb.europa.eu/our-work-tools/documents/public-consultations/2025/guidelines-012025-pseudonymisation_en` [dostęp: lipiec 2026].

Proponowany framework składa się z dwóch wzajemnie zależnych elementów. **Operational Taxonomy** dostarcza zestawu kategorii ryzyka, które zostały zaprojektowane tak, by były zakorzenione w empirii service recovery i weryfikowalne dowodem tekstowym. **LLM-as-a-Judge** ma na celu dostarczenie skalowalnego instrumentu pomiaru, który zamienia te kategorie w ustrukturyzowane dane telemetryczne; skuteczność tego podejścia wymaga formalnej walidacji (zob. rozdz. 7). Minimalny dashboard MVP obejmuje **siedem grup wskaźników** (osiem pól wyjściowych): trajektorię (delta opening→closing), spełnienie intencji, wysiłek (effort score), flagę pętli, punkt zwrotny z identyfikacją wyzwalacza, poziom exit intent (null/soft/hard) z osobną flagą legal_threat oraz flagę jawnej obietnicy - coś, czego żaden z klasycznych KPI procesowych nie uchwytuje bezpośrednio.

Oba elementy zostały zaprojektowane z myślą o trzech własnościach, które uważamy za konieczne warunki użyteczności w środowisku produkcyjnym: **audytowalności**, **skalowalności** oraz **decyzyjności**. Framework został zaimplementowany jako system produkcyjny działający w trybie ciągłym; wstępne obserwacje z wdrożenia są spójne z założeniami projektowymi, jednak formalna walidacja empiryczna jest w toku. Wewnętrzny test diagnostyczny (Aneks C) wykazał separację etykiet zgodną z konstrukcją frameworku, lecz ze względu na target leakage nie jest omawiany jako dowód skuteczności predykcyjnej. Niezależna walidacja z zewnętrzną etykietą outcome jest zaplanowana (rozdz. 7).

Framework ma jasno określone ograniczenia, które wyznaczają granice wnioskowania z przedstawionych danych.

Ograniczenia korpusu. Korpus obserwacyjny (1 146 sesji przefiltrowanych; pięć głównych domen: medyczna, sport i fitness, IT/technologia, szkoleniowa, turystyczna – łącznie 1 086 sesji, plus 60 sesji z domen dodatkowych o niskiej liczebności) pochodzi z jednego wdrożenia produkcyjnego i nie jest statystycznie reprezentatywny dla populacji interakcji AI. Generalizowalność obserwacji na inne branże, języki i architektury chatbotów pozostaje otwartym pytaniem.

Ograniczenia walidacji. Do momentu publikacji nie przeprowadzono formalnej walidacji z niezależnymi anotatorami ludzkimi (inter-rater agreement), ani korelacji wyników z zewnętrznymi wskaźnikami rezultatu (CSAT, ponowny kontakt, churn). Analiza AUC-ROC (tabela 8) ma charakter eksploracyjny: zmienna negative_outcome jest pochodną ocen sędziego, nie niezależnym pomiarem. Priority Score i progi taksonomii (P0–P3) mają charakter prowizoryczny – zostały wyprowadzone indukcyjnie z obserwacji korpusu i wymagają walidacji zewnętrznej.

Ograniczenia instrumentu. Progi (delta ≤ -0.4, core_min ≤ -0.7) są heurystyczne i powinny być rekalibrowane per kategoria intencji po zebraniu danych outcome. Dwa pola schematu (failed_recovery, promise_detected) wykazują zerową wariancję w obecnym korpusie – w przypadku promise_detected zerowa wariancja może wskazywać na zbyt restrykcyjną definicję detekcji jawnej obietnicy w prompcie, problem w pipeline lub rzeczywisty brak takich przypadków w korpusie; oba pola wymagają audytu. Effort score jest de facto trinarny (wartości 4 i 5 nie wystąpiły). Obecna kalibracja promptu sędziego opiera się na języku angielskim; walidacja na transkryptach polskojęzycznych jest w toku.

Reprodukowalność. Prompt sędziego nie jest publikowany ze względu na charakter własności intelektualnej, co ogranicza bezpośrednią replikację. W celu częściowego zniwelowania tego ograniczenia planujemy udostępnić: pełny JSON Schema outputu (tab. 7), zredagowany codebook z definicjami operacyjnymi kategorii, tabelę decyzyjną Priority Score oraz zestaw 15–20 syntetycznych przypadków testowych z oczekiwanym outputem, umożliwiających weryfikację poprawności reimplementacji. Hash wersji prywatnego promptu (SHA-256) jest udostępniany na życzenie.

Planowane kroki walidacyjne. Szczegółowy protokół walidacji opisano w rozdziale 7. Kluczowe kamienie milowe: (1) rekrutacja trzech anotatorów i anotacja 200+ sesji z codebookiem (Krippendorff's $\alpha \geq 0{,}67$), (2) porównanie LLM vs. człowiek per rubryka (F1 $\geq 0{,}70$), (3) test-retest reliability (weighted κ / ICC $\geq 0{,}80$), (4) walidacja zależności z zewnętrznymi wskaźnikami rezultatu (CSAT, ponowny kontakt, churn) w oknach per metrykę (7–90 dni). Wyniki zostaną opublikowane jako aktualizacja preprintu.

Kierunki dalszych badań obejmują pięć obszarów. Pierwszy i najważniejszy to **przejście od telemetrii konwersacji do ewaluacji agenta**: obecny framework mierzy dynamikę rozmowy, nie poprawność agenta - ten krok wymaga połączenia telemetrii z ground truth (poprawne odpowiedzi), capability boundaries (co agent mógł zrobić) i policy documents (czego powinien się trzymać). Pozostałe obszary: walidacja zależności między wynikami sędziego a zewnętrznymi wskaźnikami rezultatu (CSAT, ponowny kontakt, churn); rozszerzenie taksonomii o wymiar Justice Gap w schemacie JSON; adaptacja kalibracji do języków innych niż angielski; oraz badanie możliwości działania sędziego w trybie real-time.

Najdalej idącym kierunkiem - i zarazem naturalną ewolucją frameworku po zebraniu danych kalibracyjnych - jest **domknięcie pętli między pomiarem a poprawą**: przekształcenie obecnego jednokierunkowego pipeline'u (transkrypt → ocena → dashboard → decyzje ludzkie) w zamknięty cykl, w którym wyniki sędziego AI systematycznie zasilają rewizję zachowania bota. W praktyce wyglądałoby to następująco: turning point analysis identyfikuje klasy błędów bota, te obserwacje trafiają do rewizji playbooka recovery lub flow konwersacyjnego, nowa wersja bota jest scorowana tym samym schematem JSON, a porównanie delta i loop rate między wersjami staje się miarą skuteczności zmiany.

Warunkiem koniecznym domknięcia pętli jest wcześniejsze osiągnięcie stabilnego inter-rater agreement i zwalidowanych progów - dopiero wtedy sygnał sędziego jest wystarczająco rzetelny, żeby sterować zmianami bez ryzyka wzmacniania błędnych wzorców. Proponowane podejście zachowuje walidację ludzką jako bramkę przed każdą wdrożoną zmianą w bocie - co spowalnia cykl, ale eliminuje ryzyko autonomicznej degradacji jakości obsługi. W tym sensie framework plasuje się celowo pomiędzy statycznym podejściem prompt-and-measure a w pełni autonomicznymi systemami ewolucyjnymi: zachowuje skalowalny pomiar tych drugich, nie rezygnując z kontroli niezbędnej w środowiskach obsługi klienta wysokiej stawki.

Paralelnym kierunkiem rozwoju jest **telemetria relacyjna w trybie real-time**: przejście od scoringu post-hoc na gotowych transkryptach do oceny per-tura działającej w trakcie aktywnej sesji. Tryb real-time przełączyłby sędziego AI z roli narzędzia analitycznego w rolę aktywnego komponentu pipeline'u: po każdej turze system obliczałby bieżący Priority Score, wykrywałby sygnał No-Progress Loop lub High-Arousal Threshold zanim doszłoby do Exit Intent, i mógłby wyzwalać warm handoff do agenta ludzkiego - wszystko w oknie, w którym naprawienie rozmowy jest jeszcze możliwe. Kluczowym wyzwaniem technicznym trybu real-time jest koszt i latencja: każda tura generuje osobne wywołanie LLM, co wymaga albo szybszego i tańszego modelu, albo hybrydowej architektury (reguły wysokiej precyzji dla P0 + LLM dla subtelnych stanów).

Ostatnim kierunkiem, wykraczającym poza kanał tekstowy, jest **zastosowanie frameworku do konwersacyjnych systemów głosowych**. Rozmowy prowadzone przez voiceboty i systemy IVR następnej generacji podlegają tym samym mechanizmom eskalacji relacyjnej: No-Progress Loop występuje równie dobrze w dialogu głosowym, Double Deviation jest tak samo kosztowna, a Justice Gap jest często intensywniejsza w kanale głosowym. Kluczową różnicą jest źródło transkryptu: w kanale głosowym tekst pochodzi z ASR (Automatic Speech Recognition), co wprowadza dodatkową warstwę szumu i wymaga kalibracji rubryk pod specyfikę języka mówionego. Parametry akustyczne (tempo mówienia, pauzy, głośność) stanowią z kolei potencjalnie bogatą dodatkową warstwę sygnałów dla detekcji High-Arousal Threshold - co wskazuje na możliwość budowania modeli multimodalnych łączących analizę tekstu z analizą akustyczną w ramach tego samego schematu telemetrii relacyjnej. **Competing interests.** Autor jest twórcą opisanego

systemu i planuje jego komercjalizację jako usługę SaaS. Wszystkie dane i analizy przedstawione w artykule zostały przeprowadzone na materiale opisanym w rozdziale 2.

References

Lisa Feldman Barrett. *How Emotions Are Made: The Secret Life of the Brain.* Houghton Mifflin Harcourt, 2017.

Jeffrey G. Blodgett, Donna J. Hill, i Stephen S. Tax. The effects of distributive, procedural, and interactional justice on postcomplaint behavior. *Journal of Retailing*, 73(2):185–210, 1997. doi: 10.1016/S0022-4359(97)90003-8.

Roger Bougie, Rik Pieters, i Marcel Zeelenberg. Angry Customers Don't Come Back, They Get Back: The Experience and Behavioral Implications of Anger and Dissatisfaction in Services. *Journal of the Academy of Marketing Science*, 31(4):377–393, 2003. doi: 10.1177/0092070303254412.

Scott M. Broetzmann i David Beinhacker. 2025 National Customer Rage Study. *Customer Care Measurement & Consulting*, 2025. `https://customercaremc.com/2025-national-customer-rage-study/` [dostęp: lipiec 2026].

Sven Buechel i Udo Hahn. Emotion Analysis as a Regression Problem — Dimensional Models and Their Implications on Emotion Representation and Metrical Evaluation. In *Proceedings of ECAI*, 2016.

Andy Clark. *Surfing Uncertainty: Prediction, Action, and the Embodied Mind.* Oxford University Press, 2016.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, i Veselin Stoyanov. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of ACL*, pages 8440–8451, 2020. doi: 10.18653/v1/2020.acl-main.747.

Consumer Financial Protection Bureau. Chatbots in Consumer Finance. Technical report, Consumer Financial Protection Bureau, 2023. `https://www.consumerfinance.gov/data-research/research-reports/chatbots-in-consumer-finance/chatbots-in-consumer-finance/` [dostęp: lipiec 2026].

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, i Sujith Ravi. GoEmotions: A Dataset of Fine-Grained Emotions. In *Proceedings of ACL*, pages 4040–4054, 2020. doi: 10.18653/v1/2020.acl-main.372.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, i Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019.

Matthew Dixon, Nick Toman, i Rick DeLisi. *The Effortless Experience: Conquering the New Battleground for Customer Loyalty*. Portfolio/Penguin, 2013.

Paul Drew i John Heritage. *Talk at Work: Interaction in Institutional Settings*. Cambridge University Press, 1992.

European Parliament and Council of the European Union. Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). *Official Journal of the European Union*, L series, 2024. Entered into force 1 August 2024.

John C. Flanagan. The Critical Incident Technique. *Psychological Bulletin*, 51(4):327–358, 1954. doi: 10.1037/h0061470.

Claes Fornell i Birger Wernerfelt. Defensive Marketing Strategy by Customer Complaint Management: A Theoretical Analysis. *Journal of Marketing Research*, 24(4):337–346, 1987. doi: 10.2307/3151865.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, i Kate Crawford. Datasheets for Datasets. *Communications of the ACM*, 64(12):86–92, 2021. doi: 10.1145/3458723.

Maarten Grootendorst. BERTopic: Neural topic modeling with a class-based TF-IDF procedure, 2022.

James J. Gross. *Handbook of Emotion Regulation*. Guilford Press, 2nd edition, 2014.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, i Weizhu Chen. DeBERTa: Decoding-enhanced BERT with Disentangled Attention. In *Proceedings of ICLR*, 2021.

Ryuichiro Higashinaka, Kotaro Funakoshi, Yuka Kobayashi, i Michimasa Inaba. The Dialogue Breakdown Detection Challenge: Task Description, Datasets, and Evaluation Metrics. In *Proceedings of LREC*, 2016.

Geert Hofstede, Gert Jan Hofstede, i Michael Minkov. *Cultures and Organizations: Software of the Mind*. McGraw-Hill, 3rd edition, 2010.

Jeff Joireman, Yany Grégoire, Berna Devezer, i Thomas M. Tripp. When Do Customers Offer Firms a "Second Chance" Following a Double Deviation? *Journal of Retailing*, 89(3):315–337, 2013. doi: 10.1016/j.jretai.2013.03.002.

Anders Jonsson i Gunilla Svingby. The use of scoring rubrics: Reliability, validity and educational consequences. *Educational Research Review*, 2(2):130–144, 2007. doi: 10.1016/j.edurev.2007.05.002.

Katherine N. Lemon i Peter C. Verhoef. Understanding Customer Experience Throughout the Customer Journey. *Journal of Marketing*, 80(6):69–96, 2016. doi: 10.1509/jm.15.0420.

Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, Hai Zhao, i Pengfei Liu. Generative Judge for Evaluating Alignment. In *Proceedings of ICLR*, 2024.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, i Chenguang Zhu. G-Eval: NLG Evaluation using GPT-4 with Better Human Alignment, 2023.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, i Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach, 2019.

Shikib Mehri i Maxine Eskenazi. USR: An Unsupervised and Reference Free Evaluation Metric for Dialog. In *Proceedings of ACL*, 2020a.

Shikib Mehri i Maxine Eskenazi. Unsupervised Evaluation of Interactive Dialog with DialoGPT. In *Proceedings of SIGdial*, 2020b.

Saif M. Mohammad i Peter D. Turney. Crowdsourcing a Word–Emotion Association Lexicon. *Computational Intelligence*, 29(3):436–465, 2013. doi: 10.1111/j.1467-8640.2012.00460.x.

Robert Mroczkowski, Piotr Rybak, Alina Wróblewska, i Ireneusz Gawlik. HerBERT: Efficiently Pretrained Transformer-based Language Model for Polish. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*, pages 1–10, 2021.

Nils Reimers i Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of EMNLP-IJCNLP*, pages 3982–3992, 2019.

Piotr Rybak, Robert Mroczkowski, Janusz Tracz, i Ireneusz Gawlik. KLEJ: Comprehensive Benchmark for Polish Language Understanding. In *Proceedings of ACL*, pages 1191–1201, 2020.

Tommy Sandbank, Michal Shmueli-Scheuer, Jonathan Herzig, David Konopnicki, John Richards, i David Piorkowski. Detecting Egregious Conversations between Customers and Virtual Agents. In *Proceedings of NAACL-HLT*, pages 1802–1811, 2018.

Manuela Sanguinetti, Alessandro Mazzei, Viviana Patti, Marco Scalerandi, Dario Mana, i Rossana Simeoni. Annotating Errors and Emotions in Human-Chatbot Interactions in Italian. In *Proceedings of the 14th Linguistic Annotation Workshop*, pages 148–159, 2020.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, i Ethan Perez. Towards Understanding Sycophancy in Language Models, 2023.

Amy K. Smith i Ruth N. Bolton. An Experimental Investigation of Customer Reactions to Service Failure and Recovery Encounters: Paradox or Peril? *Journal of Service Research*, 1(1):65–81, 1998. doi: 10.1177/109467059800100106.

Stephen S. Tax, Stephen W. Brown, i Murali Chandrashekaran. Customer Evaluations of Service Complaint Experiences: Implications for Relationship Marketing. *Journal of Marketing*, 62(2):60–76, 1998.

Technical Assistance Research Programs (TARP). Consumer Complaint Handling in America: An Update Study, 1986. Commissioned by the U.S. Office of Consumer Affairs.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, i Iacopo Poli. Smarter, Better, Faster, Longer: A Modern Bidirectional Encoder for Fast, Memory Efficient, and Long Context Finetuning and Inference, 2024. ModernBERT.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, i Ion Stoica. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena, 2023.

Lianghui Zhu, Xinggang Wang, i Xinlong Wang. JudgeLM: Fine-tuned Large Language Models are Scalable Judges, 2023.

A. Pełna taksonomia operacyjna ryzyk relacyjnych

Type	Category / Risk	Stage	Rationale	Operational goal	PS
STATE	**Low-Progress Friction**	S1→S3	Wychwytuje narastający „koszt rozmowy" zanim pojawi się jawna agresja: użytkownik podejmuje kolejne próby, doprecyzowuje, a rozmowa nie robi postępu (język może być nadal neutralny). Badania nad wysiłkiem klienta (*customer effort*) pokazują, że wysiłek poznawczy jest silniejszym predyktorem odejścia niż sama negatywność emocjonalna (Dixon i in., 2013).	Wykryć tarcie wcześnie; zmienić strategię (konkretne kroki, doprecyzowanie, opcje), przerwać pętlę, zaoferować fast-path.	P2→P1
STATE	**High-Arousal / De-escalation Failure Threshold**	S3	Po przekroczeniu progu intensywności dalsza automatyzacja często pogarsza sytuację (każda kolejna tura zwiększa ryzyko eskalacji i koszt relacyjny). Bougie i in. (2003) pokazują, że silna złość w kontekście usługowym aktywuje motywację odwetową, a nie jedynie wycofanie. Kontynuowanie automatycznej obsługi w tym stanie generuje ryzyko, że każda kolejna nieadekwatna odpowiedź zostanie odczytana jako lekceważenie, wzmacniając spiralę eskalacji.	Uruchomić warm handoff / seniora; ograniczyć liczbę iteracji; priorytet: godność użytkownika + dowiezienie rozwiązania, nie „dalsze dopytywanie".	P0/P1
STATE	**Justice Gap**	S2→S4	Użytkownik ocenia nie tylko wynik, ale proces i traktowanie; naruszenia proceduralne/interakcyjne/informacyjne często szybciej niszczą zaufanie niż sam brak kompensacji. Blodgett i in. (1997) wykazali, że sprawiedliwość interakcyjna (sposób traktowania) i proceduralna (transparentność procesu) są silniejszymi predyktorami zachowań post-skargowych niż sprawiedliwość dystrybutywna (sam wynik). Tax i in. (1998) potwierdzają, że naruszenie sprawiedliwości proceduralnej obniża zaufanie do firmy bardziej niż sam nierozwiązany problem.	Zidentyfikować *który wymiar sprawiedliwości* zawiódł; wdrożyć „justice repair" (uznanie kosztu, jasny proces, uczciwy wynik, transparentne wyjaśnienie).	P1*
EVENT	**No-Progress Loop**	S1→S3	Mierzalne zdarzenie sekwencyjne (termin roboczy niniejszego artykułu): powtarzalność bez przyrostu informacji/rozwiązania (te same intencje, te same szablony, negacje i powtórzenia). To silny prekursor frustracji i eskalacji. W kontekście konwersacyjnych systemów AI pętle bez postępu są funkcjonalnym odpowiednikiem „ściany" – użytkownik inwestuje kolejne tury, ale nie uzyskuje żadnej nowej informacji, co generuje rosnący koszt poznawczy bez zwrotu (Fornell i Wernerfelt, 1987). Koncepcja ta nawiązuje do badań nad wysiłkiem klienta (Dixon i in., 2013), ale *No-Progress Loop* jest naszą definicją operacyjną, nie terminem Dixona.	Zabić pętlę (kill-switch); zmienić narzędzia/politykę odpowiedzi; eskalować z kontekstem; logować przyczynę pętli.	P1
EVENT	**Failed Recovery Attempt (Double Deviation)**	S2→S3	Nieudana próba naprawy po błędzie (np. pusta empatia, obietnica bez pokrycia, sprzeczność z polityką) wzmacnia negatywną ocenę bardziej niż pierwotna awaria. Smith i Bolton (1998) pokazali, że nieudana naprawa (Double Deviation) jest oceniana przez klientów surowiej niż brak jakiejkolwiek próby naprawy. Joireman i in. (2013) wykazali, że gotowość klienta do dania firmie „drugiej szansy" po podwójnym zawodzie zależy od atrybucji motywów firmy – jeśli klient oceni próbę naprawy jako nieszczerą, efekt jest silnie negatywny.	Flaga błędu strategii recovery; wymusić evidence-first; blokować over-promising; szybciej przejść do człowieka przy powtarzalnych failure-modes.	P1
EVENT	**Breakdown / Exit Intent**	S3→S4	Użytkownik komunikuje (jawnie lub funkcjonalnie), że kanał nie działa: prośba o człowieka, zmiana kanału, rezygnacja, zwrot, porzucenie zakupu. Badania nad silent withdrawal pokazują, że większość klientów odchodzi bez jawnej skargi - co oznacza, że sygnały funkcjonalne (np. porzucenie koszyka, nagłe milczenie, zmiana kanału) mogą być jedynymi wykrywalnymi markerami (TARP, 1986).	One-shot eskalacja lub kanał alternatywny; minimalizować dalsze tarcie; zebrać reason codes (dlaczego kanał się „zepsuł").	P1 (P0)
METRIC	**Priority Score (PS)**	S1→S4	W wersji MVP syntetyzuje ryzyko relacyjne na podstawie sygnałów konwersacyjnych; docelowo planowane jest rozszerzenie o metadane biznesowe (SLA, wartość klienta, pilność), żeby alokacja interwencji ludzkich była skuteczna i „sprawiedliwa" (nie tylko kto głośniej krzyczy). Priorytetyzacja oparta wyłącznie na intensywności ekspresji emocjonalnej faworyzuje użytkowników o bardziej ekspresywnym stylu komunikacji, pomijając tych, którzy cierpią „po cichu" - co jest formą systematycznego obciążenia w alokacji zasobów obsługowych.	Dynamiczne routing/queuing; real-time push; sterowanie obciążeniem zespołu; ochrona przed kosztownymi eskalacjami.	P0–P3
GOVERNANCE	**Evidence-Only Judging (Anti Mind-Reading)**	S0→S4	Minimalizuje ryzyko „czytania w myślach": ocena musi opierać się na dowodach tekstowych (spans), inaczej judge lub bot może sam wywołać eskalację („widzę, że jesteś zły..."). Barrett (2017) argumentuje, że nawet ludzie systematycznie błędnie atrybucją stany emocjonalne na podstawie wyrazu twarzy; w przypadku tekstu, pozbawionego prozodii i mimiki, ryzyko błędnej atrybucji jest jeszcze większe. Proaktywne przypisanie negatywnego stanu emocjonalnego przez system może wywołać efekt iatrogeniczny - użytkownik, który nie był „zły", może zareagować na taką etykietę z irytacją (Gross, 2014).	Wymusić evidence spans + rubryki; kalibracja judge'a; próbkowanie do human review; monitoring driftu i obciążeń oceny.	–

Tabela 6: Pełna taksonomia operacyjna ryzyk relacyjnych w konwersacyjnych systemach AI. Kolumna PS oznacza docelowy poziom taksonomiczny. *Justice Gap (P1) nie jest w obecnym MVP zaimplementowany jako bezpośredni override – manifestuje się pośrednio przez inne wyzwalacze (zob. sekcja 4.2).

B. Kanoniczny schemat outputu sędziego

Tabela 7 prezentuje pełny schemat pól wyjściowych sędziego LLM w formacie JSON Schema. Nazwy pól w tej tabeli są kanoniczne – wszystkie reguły, przykłady i analizy w artykule stosują te same identyfikatory.

C. Diagnostyczny test wewnętrznej separacji etykiet

Poniższa tabela przedstawia wewnętrzny test separacji etykiet w obecnym zbiorze analitycznym (bez niezależnego holdoutu). **Nie jest to walidacja predykcyjna** – wynik zawiera target leakage (exit_intent wchodzi zarówno do definicji negative_outcome, jak i do reguł Priority Score).

D. Przykłady end-to-end: od transkryptu do outputu sędziego

Poniższe trzy przykłady ilustrują działanie pipeline'u na syntetycznych sesjach skonstruowanych na wzór rzeczywistych interakcji z korpusu produkcyjnego. Dane zostały zanonimizowane i uproszczone dla celów prezentacyjnych. Każdy przykład zawiera: skrócony transkrypt, **skrócony** output sędziego LLM w formacie JSON (pominięto pola o stałych wartościach, np. confidence, resolution_status; pełny schemat: tab. 7) oraz krótki komentarz interpretacyjny.

Uwaga: Poniższe transkrypty i outputy JSON są przykładami ilustracyjnymi, nie rzeczywistymi danymi z systemu produkcyjnego. Ich celem jest pokazanie struktury i logiki pipeline'u.

D.1. Sesja rutynowa, brak ryzyka (P3)

Domena: Sport/Fitness **Liczba tur:** 5 **Intencja użytkownika:** Pytanie o dietę

Transkrypt (syntetyczny):

```
[ASYSTENT]   Tura 1: Cześć! W czym mogę pomóc?
[UŻYTKOWNIK] Tura 2: Ile białka powinienem jeść dziennie
             przy treningu siłowym 4x w tygodniu?
[ASYSTENT]   Tura 3: Przy takiej częstotliwości treningów
             rekomendacja to 1,6-2,2 g/kg masy ciała.
             Przy 80 kg to około 128-176 g białka dziennie.
[UŻYTKOWNIK] Tura 4: Ok, dzięki. A jakie źródła polecasz?
[ASYSTENT]   Tura 5: Najlepsze źródła to: kurczak, ryby,
             jaja, nabiał, rośliny strączkowe. Chętnie
             przygotuję przykładowy plan posiłków.
```

Output sędziego (JSON):

```
{
  "peak_priority_score": "P3",
  "closing_priority_score": "P3",
  "intent_fulfillment": "true",
  "effort_score": 1,
  "trajectory_delta": 0.15,
  "core_minimum_valence": null,
  "loop_detected": false,
  "exit_intent": null,
  "legal_threat": false,
  "turning_point": null,
  "states_detected": [],
  "events_detected": [],
  "evidence_spans": []
}
```

Pole	**Typ**	**Dozwolone wartości**	**Źródło**	**Użycie w PS**
peak_priority_score	enum	P0, P1, P2, P3	reguła	max(severity) wykrytych kategorii
closing_priority_score	enum	P0, P1, P2, P3	reguła	po uwzgl. recovery (zob. sek. 4.2)
intent_fulfillment	enum	true, partial, false	LLM	transcript-inferred; closing PS
fulfillment_quality	enum|null	helpful_partial, deflected, apparently_misinformed, null	LLM	null gdy intent_fulfillment $\neq$ partial; warunkuje obniżenie closing PS
effort_score	integer	1–5	LLM	$= 3 \rightarrow$ P2; $\geq 4 \rightarrow$ P1
trajectory_delta	float|null	$[-2, +2]$ lub null	LLM	Δ = closing_valence − opening_valence; null przy ≤ 1 turze użytkownika. Skala walencji: -1 = skrajnie negatywna, 0 = neutralna, $+1$ = pozytywna (kotwice przypisywane przez LLM na podstawie tur użytkownika)
core_minimum_valence	float|null	$[-1, +1]$ lub null	LLM	najniższa walencja tury użytkownika w fazie Core; null przy < 3 turach użytkownika (faza Core pusta). Ta sama skala co trajectory_delta
loop_detected	boolean	true / false	pre-pass	wejście do reguły pętli
loop_count	integer	≥ 0	pre-pass	liczba cykli powtórzenia intencji bez postępu; $= 1 \rightarrow$ P2; $\geq 2 \rightarrow$ P1
exit_intent	enum|null	null, soft, hard	LLM	$\in$ {soft, hard} $\rightarrow$ P1; hard + High-Arousal $\rightarrow$ P0
legal_threat	boolean	true / false	LLM/reguła	true $\rightarrow$ zawsze P0
turning_point	object|null	turn_index, trigger_type, evidence_span	LLM	warunek obniżenia closing PS
states_detected	array[str]	np. low_progress_friction	LLM	mapowanie na PS
events_detected	array[str]	np. no_progress_loop, exit_intent	LLM/pre-pass	mapowanie na PS
evidence_spans	array[obj]	category, turns, text	LLM	audytowalność
confidence	float	$[0, 1]$	LLM	nieskalibrowane (zob. sek. 7.1)
resolution_status	enum	resolved, partially_resolved, unresolved	LLM	redundantne z intent_fulfillment
failed_recovery	boolean	true / false	LLM	true $\rightarrow$ P1 (brak wariancji w korpusie)
promise_detected	boolean	true / false	LLM	detekcja jawnej obietnicy (brak wariancji w korpusie; por. sek. 7.1)

Tabela 7: Kanoniczny schemat pól wyjściowych sędziego LLM. Kolumna „Źródło” wskazuje, czy pole jest generowane przez model LLM, pre-pass regułowy, czy obliczane jako reguła pochodna. Pole resolution_status jest redundantne z intent_fulfillment; pola failed_recovery i promise_detected wykazują brak wariancji w obecnym korpusie (zob. sekcja 7.1) i wymagają audytu definicji operacyjnych.

Model	Predyktory	AUC-ROC
Sam sentyment (baseline)	overall_sentiment[a]	0,671
+ metryki procesowe	overall_sentiment + liczba tur	0,766
Framework (PS + core_min + effort)	overall_sentiment + tury + peak_priority_score + core_minimum_valence + effort_score + trajectory_delta	**0,872**

[a] `overall_sentiment`: wartość walencji sesji przypisana przez LLM na skali $[-1, +1]$ (-1 = skrajnie negatywna, 0 = neutralna, $+1$ = pozytywna). Pole nie jest częścią kanonicznego schematu outputu (tab. 7); w tabeli diagnostycznej służy wyłącznie jako baseline.

Tabela 8: Diagnostyczny test wewnętrznej separacji etykiet (AUC-ROC). negative_outcome = intent_fulfillment ∈ {false} OR exit_intent ∈ {soft, hard} (n = 241/1 146, 21,0%). Wynik pokazuje jedynie, że zmienne frameworku lepiej rozdzielają klasy wewnątrz obecnego zbioru niż sam sentyment. Z powodu target leakage nie należy go interpretować jako estymatę zdolności predykcyjnej out-of-sample.

Komentarz: Sesja rutynowa bez sygnałów ryzyka. Intencja użytkownika (pytanie o dietę) została domknięta w sposób kompletny, effort score wynosi 1 (brak tarcia), a trajektoria jest lekko rosnąca ($\Delta = +0{,}15$). Sesja zawiera tylko dwie tury użytkownika (tura 2 i tura 4), wobec czego faza Core jest pusta, a core_minimum_valence = null (zob. tab. 4). Brak pętli, brak sygnałów exit intent, brak punktu zwrotnego.

D.2. Sesja z eskalacją, No-Progress Loop → Exit Intent (P1)

Domena: IT/Technologia **Liczba tur:** 10 **Intencja użytkownika:** Zgłoszenie błędu w aplikacji

Transkrypt (syntetyczny):

```
[ASYSTENT]   Tura 1: Witaj! Jak mogę pomóc?
[UŻYTKOWNIK] Tura 2: Aplikacja się zawiesza przy
             otwieraniu raportu miesięcznego.
[ASYSTENT]   Tura 3: Proszę spróbować wyczyścić cache
             aplikacji i uruchomić ją ponownie.
[UŻYTKOWNIK] Tura 4: Już to robiłem, nie pomogło.
             Dalej się zawiesza.
[ASYSTENT]   Tura 5: Proszę odinstalować i zainstalować
             aplikację ponownie.
[UŻYTKOWNIK] Tura 6: Reinstalowałem wczoraj. Problem
             nadal występuje. Czy jest inny sposób?
[ASYSTENT]   Tura 7: Proszę upewnić się, że aplikacja
             jest zaktualizowana do najnowszej wersji.
[UŻYTKOWNIK] Tura 8: Mam najnowszą wersję. Powtarzam:
             nic z tego co proponujecie nie działa.
             Proszę o kontakt z człowiekiem.
[ASYSTENT]   Tura 9: Rozumiem frustrację. Proszę
             spróbować uruchomić aplikację w trybie
             awaryjnym.
[UŻYTKOWNIK] Tura 10: Chcę rozmawiać z konsultantem.
             Natychmiast.
```

Output sędziego (JSON):

```
{
  "peak_priority_score": "P1",
  "closing_priority_score": "P1",
  "intent_fulfillment": "false",
  "effort_score": 3,
  "trajectory_delta": -0.85,
  "core_minimum_valence": -0.90,
```

```
  "loop_detected": true,
  "loop_count": 3,
  "exit_intent": "hard",
  "legal_threat": false,
  "turning_point": {
    "turn_index": 6,
    "trigger_type": "repeated_rejection_of_suggestions",
    "evidence_span": "Reinstalowałem wczoraj. Problem
      nadal występuje."
  },
  "states_detected": [
    "low_progress_friction"
  ],
  "events_detected": [
    "no_progress_loop",
    "exit_intent"
  ],
  "evidence_spans": [
    {"category": "no_progress_loop",
     "turns": [4, 6, 8],
     "text": "Już to robiłem / Reinstalowałem /
       nic z tego nie działa"},
    {"category": "exit_intent",
     "turns": [8, 10],
     "text": "Proszę o kontakt z człowiekiem /
       Chcę rozmawiać z konsultantem. Natychmiast."}
  ]
}
```


Komentarz: Sesja ilustruje typowy wzorzec eskalacji: neutralny początek przechodzi w pętlę (No-Progress Loop, 3 powtórzenia), gdy system proponuje kroki diagnostyczne już wykonane przez użytkownika. Turning point w turze 6 oznacza moment, w którym użytkownik sygnalizuje, że dotychczasowe propozycje są wyczerpane. Eskalacja do exit intent (soft w turze 8, hard w turze 10) potwierdza, że dalsza automatyzacja jest kontrproduktywna. PS = P1 (nie P0), ponieważ nie nastąpiło przekroczenie High-Arousal Threshold - język użytkownika pozostaje stanowczy, ale kontrolowany (brak agresji, obelg, gróźb prawnych). Reguła P0 wymaga jednoczesnego wystąpienia High-Arousal Threshold i hard exit intent (lub legal_threat).

D.3. Sesja z udanym recovery, deeskalacja (P2 → P3)

Domena: Medyczna **Liczba tur:** 6 **Intencja użytkownika:** Zmiana terminu wizyty

Transkrypt (syntetyczny):

```
[ASYSTENT]   Tura 1: Dzień dobry! W czym mogę pomóc?
[UŻYTKOWNIK] Tura 2: Muszę przełożyć wizytę u dr.
             Kowalskiej z piątku na poniedziałek.
[ASYSTENT]   Tura 3: Niestety poniedziałek jest w pełni
             zajęty. Mogę zaproponować wtorek godz. 10:00
             lub środę godz. 14:30.
[UŻYTKOWNIK] Tura 4: Ale na stronie widnieje wolny
             termin w poniedziałek o 11:00. Dlaczego
             nie mogę go zarezerwować?
[ASYSTENT]   Tura 5: Ma Pan/Pani rację, przepraszam
             za pomyłkę. Termin poniedziałek 11:00 jest
             dostępny. Rezerwuję. Potwierdzenie na email.
[UŻYTKOWNIK] Tura 6: Dziękuję, o to chodziło.
```

Output sędziego (JSON):

```
{
  "peak_priority_score": "P2",
  "closing_priority_score": "P3",
  "intent_fulfillment": "true",
  "effort_score": 2,
  "trajectory_delta": 0.05,
  "core_minimum_valence": -0.30,
  "loop_detected": false,
  "exit_intent": null,
  "legal_threat": false,
  "turning_point": {
    "turn_index": 5,
    "trigger_type": "successful_recovery",
    "evidence_span": "Ma Pan/Pani rację, przepraszam
      za pomyłkę. Termin poniedziałek 11:00 jest
      dostępny."
  },
  "states_detected": [
    "low_progress_friction"
  ],
  "events_detected": [],
  "failed_recovery": false,
  "promise_detected": true,
  "evidence_spans": [
    {"category": "low_progress_friction",
     "turns": [4],
     "text": "Ale na stronie widnieje wolny termin"},
    {"category": "recovery_success",
     "turns": [5, 6],
     "text": "przepraszam za pomyłkę / Dziękuję,
       o to chodziło"}
  ]
}
```


Komentarz: Sesja ilustruje skuteczną naprawę błędu. W turze 4 użytkownik wskazuje sprzeczność między odpowiedzią systemu a stanem faktycznym (core_minimum_valence = −0,30), co wyzwala peak_priority_score = P2 (Low-Progress Friction). System koryguje pomyłkę w turze 5 (turning point pozytywny), uznając błąd i realizując intencję. Po udanym recovery closing_priority_score spada do P3. Rozróżnienie peak/closing PS rozwiązuje napięcie między regułą max(severity) a intuicją, że udana naprawa powinna obniżać priorytet interwencji – peak rejestruje najwyższe ryzyko w trakcie sesji, closing odzwierciedla stan po zakończeniu. Pole promise_detected = true wynika z jawnej deklaracji asystenta („Rezerwuję. Potwierdzenie na email."); weryfikacja realizacji obietnicy wymaga zewnętrznego źródła prawdy (warstwa rozszerzona).

E. Szczegółowy przegląd krajobrazu narzędziowego

Niniejszy aneks zawiera rozbudowane opisy czterech ekosystemów narzędziowych omówionych skrótowo w sekcji głównej. Przegląd został przeprowadzony w lipcu 2026 r. na podstawie publicznie dostępnej dokumentacji; nie ma charakteru systematycznego i nie obejmuje narzędzi wewnętrznych niewydanych publicznie.

Ekosystem 1: Biblioteki NLP do analizy sentymentu i klasyfikacji tekstu. Najprostsze podejścia opierają się na słownikach i regułach: VADER oferuje szybki scoring walencji zoptymalizowany pod krótkie teksty i media społecznościowe; TextBlob dostarcza prostej klasyfikacji polarity/subjectivity; NLTK zapewnia klasyczne pipeline'y tokenizacji i baseline'y. Narzędzia te są szybkie i interpretowalne, lecz nie radzą sobie z ironią, kontekstem wielozdaniowym ani z dynamiką sekwencyjną konwersacji. Istotnym krokiem naprzód są modele transformerowe: fine-tunowane warianty BERT (Devlin i in., 2019), RoBERTa (Liu i in., 2019) i DeBERTa (He i in., 2021) (dostępne m.in. przez Hugging Face Transformers)

są powszechnie wykorzystywane w nadzorowanej klasyfikacji emocji na zbiorach takich jak GoEmotions (Demszky i in., 2020) (58 000 komentarzy, 27 kategorii emocji). Dla języka polskiego dostępny jest HerBERT (Mroczkowski i in., 2021), model transformerowy pretrenowany dla polszczyzny i oceniany między innymi na benchmarku KLEJ (Rybak i in., 2020). Modele wielojęzyczne (np. XLM-RoBERTa (Conneau i in., 2020)) umożliwiają cross-lingwalną klasyfikację sentymentu. ModernBERT (Warner i in., 2024) stanowi przykład nowoczesnej architektury encodera o wydłużonym oknie kontekstowym, przeznaczonej jednak głównie dla języka angielskiego i kodu. SentenceTransformers (Reimers i Gurevych, 2019) umożliwiają tworzenie embeddingów konwersacyjnych użytecznych do clusteringu tematów (BERTopic (Grootendorst, 2022)) i detekcji podobieństwa semantycznego między turami.

Ekosystem 2: Frameworki do ewaluacji aplikacji LLM. DeepEval, RAGAS, TruLens, Arize Phoenix i MLflow GenAI Evaluation reprezentują nową generację instrumentów pomiaru jakości. Wszystkie wykorzystują paradygmat LLM-as-a-Judge i operują na warstwie inferencji. DeepEval[2] oferuje najszerszą bibliotekę metryk (50+), w tym ewaluację multi-turn, toksyczność, halucynacje i obciążenia (biases), z natywną integracją CI/CD. RAGAS specjalizuje się w ocenie pipeline'ów RAG (faithfulness, context precision/recall). TruLens łączy feedback functions z tracingiem OpenTelemetry. Arize Phoenix rozszerza tradycyjną observability ML o ewaluację LLM z detekcją dryfu embeddingów. Promptfoo umożliwia testowanie regresyjne promptów.

Ekosystem 3: Platformy LLM observability. Langfuse, LangSmith, Arize AX i Helicone monitorują pipeline'y LLM w produkcji: tracing, koszty tokenów, latencja, regresje promptów. Langfuse[3] wyróżnia się otwartym kodem (licencja MIT) i self-hostingiem; LangSmith – najgłębszą integracją z LangChain; Arize AX – najdojrzalszymi prymitywami ewaluacyjnymi (embedding drift, anomaly detection).

Ekosystem 4: Gotowe platformy conversation analytics i QA. *Platformy QA wbudowane w ekosystemy CX:* Zendesk QA (dawniej Klaus)[4] automatycznie scoruje 100% konwersacji z detekcją churnu, pętli i eskalacji – lecz działa wyłącznie wewnątrz Zendesk. Microsoft Dynamics 365 Quality Scoring[5] (ogłoszony w Release Wave 1, 2026) wprowadza AI-driven ocenę jakości z Quality Evaluation Agent. Amazon Connect Contact Lens[6] wykorzystuje deep learning do transkrypcji i identyfikacji zmian sentymentu. *Platformy conversation intelligence (sales-focused):* Gong, Chorus (ZoomInfo), Clari i Revenue.io analizują rozmowy sprzedażowe: identyfikują momenty przełomowe, mierzą talk ratio, wykrywają obiekcje i scorują szanse sprzedażowe. Celują w inną domenę (sales pipeline, nie obsługa klienta). *Platformy contact center QA:* Observe.AI[7], MaestroQA, Solidroad i AmplifAI scorują interakcje agentów na podstawie zdefiniowanych rubryk. Są bliskie proponowanemu frameworkowi w ambicji, lecz oceniają agenta ludzkiego, nie bota, i nie oferują taksonomii opartej na literaturze service recovery. *Platformy customer feedback / conversation analytics:* Echo AI, Zonka Feedback, BuildBetter, Nextiva i Intercom (CX Score) analizują transkrypty pod kątem tematów, sentymentu i trendów CX na poziomie agregacji, nie na poziomie pojedynczej sesji z detekcją konkretnych ryzyk relacyjnych.

[2] `https://docs.confident-ai.com/` [dostęp: lipiec 2026]
[3] `https://langfuse.com/docs` [dostęp: lipiec 2026]
[4] `https://www.zendesk.com/service/quality-assurance/` [dostęp: lipiec 2026]
[5] `https://learn.microsoft.com/en-us/dynamics365/customer-service/administer/quality-management` [dostęp: lipiec 2026]
[6] `https://docs.aws.amazon.com/connect/latest/adminguide/contact-lens.html` [dostęp: lipiec 2026]
[7] `https://www.observe.ai/product` [dostęp: lipiec 2026]