Actionable Interpretability for Churn Classification: A Text Bottleneck Model Case Study at a Major Telecom Provider

Adrian Sauter[1,*] **Vera Neplenbroek**[1,†] **Giorgos Vlassopoulos**[2] **Gianluigi Bardelloni**[2]

[1]University of Amsterdam [2]KPN

† Corresponding author: v.e.neplenbroek@uva.nl

Abstract

In subscription-based businesses, understanding why a customer intends to churn is as vital as the classification itself. We present a case study at a large European telecommunications provider, where we implement Text Bottleneck Models (TBMs) for post-call churn classification. The TBM distills dialogues into a sparse set of human-interpretable concepts and provides faithful, snippet-based evidence for every decision. We show that the TBM performs competitively with black-box baselines and demonstrate potential business impact via automated call profiling and an interactive stakeholder dashboard. Our work demonstrates that the perceived trade-off between interpretability and predictive performance can be bridged, providing the high-accuracy evidence needed for industrial retention strategies.

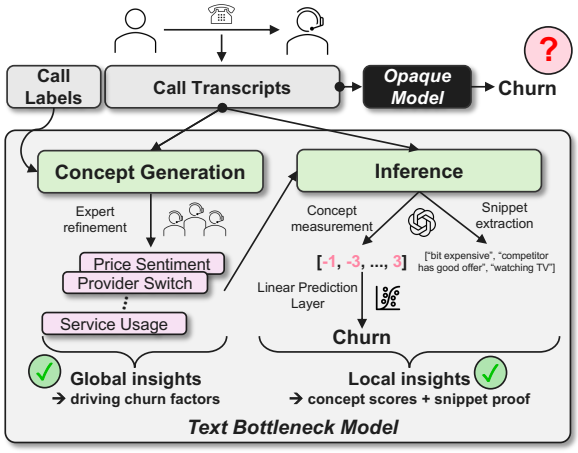

Figure 1: Comparison of a black-box text classifier (top-right) and the interpretable-by-design Text Bottleneck Model (bottom), which classifies churn through intermediate, human-understandable, actionable concepts.

1 Introduction

Managing customer churn, the full or partial cancellation of subscriptions, is vital for telecommunications providers, yet the most valuable signals are often locked within millions of minutes of unstructured customer service dialogues. While Large Language Models (LLMs) can automate the classification of churn-intent at scale, their black-box nature creates a significant barrier to industrial adoption. A binary label of "Churn" provides no guidance for retention; without knowing if a customer is frustrated by pricing, network quality, or a competitor's offer, stakeholders cannot take meaningful action. This lack of transparency forces a difficult choice between high-accuracy automated systems and the actionable insights of manual review.

We attempt to resolve this tension by implementing the Text Bottleneck Model (TBM) (Ludan et al., 2024) for post-call churn classification at a large European provider. Unlike standard classifiers, TBMs are interpretable-by-design. They map complex, multi-turn dialogues into a sparse layer of human-understandable concepts, such as *Price Sentiment* or *Service Dissatisfaction*, before making a final classification (see Figure 1). By grounding every decision in specific conversational snippets and meaningful concepts, the TBM transforms opaque model outputs into transparent insights for actionable business intervention.

Our contributions focus on the industrial application and operational utility of TBMs:

- We present a framework that extends the original TBM by combining LLM-based concept discovery with expert-in-the-loop refinement, ensuring that the resulting bottleneck is actionable and business-aligned.
- We demonstrate that the expert-refined TBM achieves predictive performance on par with black-box models, proving that transparency does not require a sacrifice in predictive power.

* Work conducted during an industry internship.

- We show how concept-level activations enable granular call profiling and present an interactive stakeholder dashboard, providing non-technical managers with evidence-based insights unavailable through binary classification.

2 Related Work

Explainability in NLP has received increasing attention alongside advances in LLMs (see Zhao et al. (2024) for an overview). A key distinction exists between *intrinsic* methods, which build interpretability into the model, and *post-hoc* explanations. Intrinsic methods include rationale-based models (Lei et al., 2016; Bastings et al., 2019) and attention-based explanations, though the faithfulness of the latter is debated (Jacovi and Goldberg, 2020; Jain and Wallace, 2019). Post-hoc methods such as LIME (Ribeiro et al., 2016) and SHAP (Lundberg and Lee, 2017) provide local, model-agnostic explanations, but often lack faithfulness and stability (Jacovi and Goldberg, 2020; Adebayo et al., 2018). In contrast, concept-based models are interpretable by design, providing global interpretability through human-understandable concepts and local interpretability via concept activations for individual samples (Koh et al., 2020).

Concept-based models aim to replace raw inputs with human-interpretable, task-relevant concepts (Koh et al., 2020). In NLP, two main directions have emerged: probing latent concepts inside pre-trained models (Sheng and Uthus, 2020; Huang et al., 2025), and designing models that predict directly through an explicit concept layer. The Text Bottleneck Model (TBM) framework (Ludan et al., 2024) follows the latter approach, together with works such as SELF-EXPLAIN (Rajagopal et al., 2021), C^3M (Tan et al., 2024), and CB-LLM (Sun et al., 2025). Compared to these, TBMs restrict the concept set to a small and stable collection, allow iterative refinement without relying on predefined concepts, and provide snippet-based evidence for each activation. This combination makes TBMs particularly suited to domains where interpretability and business actionability are as important as predictive performance, motivating our choice for the TBM in this work.

LLM prompting for labeling and explanation has gained traction as LLMs show strong zero- and few-shot classification abilities (Brown et al., 2020). Recent work demonstrates that LLMs can act as effective labelers across tasks (Zhou et al., 2023; Liang et al., 2023), though their reliability is debated (Koo et al., 2024; Bavaresco et al., 2025). In the TBM framework, however, labeling is comparatively simple: Once concepts are well defined, LLMs are expected to apply them reliably, as shown in Ludan et al. (2024).

3 Methodology

We adopt the **Text Bottleneck Model** (TBM) (Ludan et al., 2024), an interpretable alternative to black-box classifiers. The TBM introduces an intermediate layer of *business-relevant concepts* that mediate between raw transcripts and the churn classification (see Figure 2). The model consists of three stages: concept generation, concept measurement, and prediction. A formal description of the process can be found in Appendix A.

Concepts. Concepts are defined as structured JSON objects with a name, description, question, possible responses, and a response-to-score mapping (see Table 2 in Appendix F.2 for an example).

Concept Generation. Concepts can be obtained automatically via LLM prompting (all prompts can be found in Appendix M), or manually, via domain-specific refinement. In the automated setting, the LLM iteratively proposes candidate concepts aimed at separating the examples that the model misclassified in the previous iteration. These concepts are then automatically refined via another prompt, but their representation in JSON-format also allows for simple manual manipulation and refinement. While prior work has focused solely on evaluating automatically generated concepts (Ludan et al., 2024), we also explore how domain-specific refinement with experts affects prediction performance. Crucially, the TBM's modularity allows for the seamless addition, removal, or manual refinement of concepts as business requirements evolve or new data distributions emerge, ensuring the model remains performant and relevant over time.

Concept Measurement. Given a transcript, an LLM is prompted with each concept definition to assign a response from the pre-defined set of responses, which is mapped to a numerical score. The LLM also extracts short snippets as evidence, providing transparency into why a concept was activated. The result is a concept vector $\mathbf{s}(t)$ that represents the transcript in the interpretable concept bottleneck space.

Prediction Layer. Finally, a white-box classifier

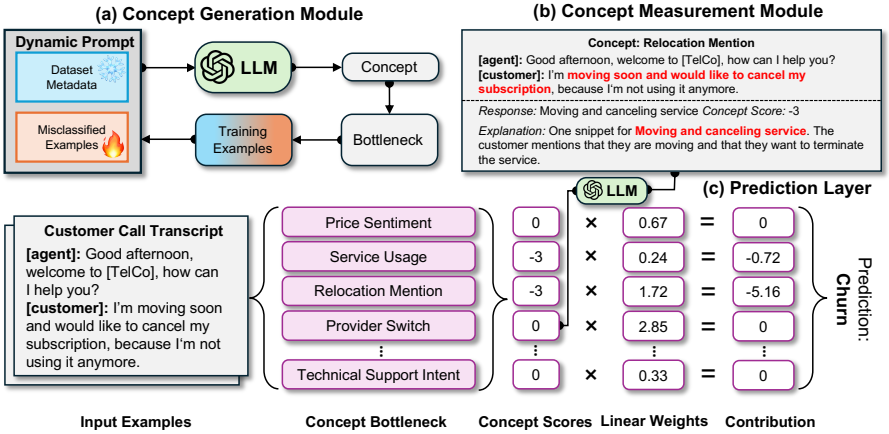

Figure 2: Structure of the Text Bottleneck Model (Figure adopted from Ludan et al. (2024)). Given an input example, the (a) Concept Generation Module iteratively discovers new concepts (e.g., *Provider Switch*). The (b) Concept Measurement Module then measures the response of each concept and extracts relevant snippets (e.g., "moving soon and would like to cancel my service") and provides a numerical concept score (e.g., -3). Finally, the (c) Prediction Layer aggregates all concept scores for the input and learns their relative weights to make the final churn prediction.

(e.g., logistic regression) is trained on the concept vectors to predict the final label, which is Churn/ No Churn in our case. The learned weights indicate the global importance of each concept, while the concept scores and evidence snippets obtained in the concept measurement step offer local explanations.

4 Experimental Setup

4.1 Dataset

Thousands of customer calls are handled daily, but the lack of labeled data prevents large-scale supervised training. We therefore use an expert-annotated dataset, where each transcript was labeled by five annotators, with final labels assigned by majority vote.

The dataset follows a 14-class schema[1], where "Cancel subscription" corresponds to "Churn" and the remaining 13 classes (e.g., "Help with malfunctions", "Technician appointment") are grouped as "No Churn." Importantly, cases where customers cancel or downgrade only parts of their subscription are also treated as Churn. We sample 200 training and 200 test transcripts, with the training set being balanced across classes (100 Churn, 100 No Churn) and the test set using the original class imbalance (28 Churn, 172 No Churn). On average, transcripts contain 270 words and 20 conversational turns between customer and agent. Due to privacy constraints, all explicit snippets and transcripts in this paper are synthetic and do not originate from real data.

4.2 Large Language Models

In our experiments, we use OpenAI's GPT-4 family of models via the Azure OpenAI API (OpenAI, 2024). Concept generation and measurement rely on `gpt-4-turbo-128k`, but we also test `gpt-4o-mini` as a lightweight alternative. Further model details and selection rationale are provided in Appendix B. We compare the TBM against two black-box models: a `BERTje`-based model (de Vries et al., 2019), which has been domain-adapted using 1.5 million dialogues and processes the first 512 tokens of a transcript to output one of the 14 previously mentioned classes, which we map to a binary Churn/ No Churn label; and a GPT-based black-box model, using either `gpt-4-turbo-128k` or `gpt-4o-mini`, which directly predicts Churn/ No Churn from the full transcript via a tailored prompt (see Appendix 11).

5 Results

This section analyzes the TBM framework from various points of view: Concept generation and prediction performance (Section 5.1), qualitative error analysis (Section 5.2), as well as exploratory analyzes using concept-based clustering (Section 5.3). We also present a practical dashboard (Section 5.4) and include additional evaluations of the robustness of concept measurements (Appendix G), a chain-of-thought ablation (Appendix H), and an analysis

[1]All datasets were preprocessed and linguistically normalized for research purposes.

of the snippet quality and sufficiency (Appendix I).

5.1 Concept Generation and Prediction Performance

Automated Concept Generation. We run three independent TBM concept generation runs using logistic regression as the prediction layer. Following Ludan et al. (2024), we set the concept acceptance threshold to $\gamma = 0.1$, meaning a newly proposed concept is retained only if it improves prediction accuracy on the training set by at least 10% relative to the previous iteration. Figure 3 visualizes the evolution of the first 8 concepts in each run, with cumulative concepts on the y-axis and test accuracy on the x-axis. Each node represents an added concept, sized by its final absolute weight in the prediction layer.

Across runs, the LLM successfully proposes semantically meaningful but diverse concepts, such as *Price Sentiment*, *Service Utilization Sentiment*, and *Communication Clarity*, resulting in final test accuracies based on 8 concepts between 0.64 and 0.72. We attribute this variability to the higher complexity of multi-turn customer calls, which differ substantially from the short, focused texts used in the original TBM experiments (e.g., the CEBaB dataset (Abraham et al., 2022), which contains restaurant reviews), where the TBM with automatically generated concepts performed on par with black-box models.

We also note that not all discovered concepts are equally actionable from a business standpoint. For instance, *Value Perception* provides clearer retention insights than abstract notions like *Communication Clarity*. This reinforces the need for human-in-the-loop refinement to ensure that automatically generated concepts align with the organization's operational objectives.

Domain Expert Refinement. To ensure domain relevance and business actionability, we collaborate with internal churn experts to manually refine the automatically generated concepts. During refinement, we focus on removing redundancy, clarifying ambiguous definitions, and prioritising factors that could meaningfully inform retention strategy. This process results in seven high-quality concepts: *Price Sentiment*, *Provider Switch*, *Relocation Mention*, *Service Modification Intent*, *Service Usage*, *Subscription Flexibility Concern*, and *Technical Support Intent*. More details on how this set of concepts was obtained, as well as the redacted concept definitions can be found in Appendix F. These concepts capture a broad range of customer motivations, from satisfaction with pricing and flexibility to service continuity intentions. Each follows a consistent response mapping from -3 (strong negative signal) to +3 (strong positive signal), ensuring comparability across the concept space.

Prediction Performance. We compare the refined TBM to two black-box baselines: the production `BERTje`-based classifier (de Vries et al., 2019), and a GPT-4 zero-shot model. We also test several white-box models in the TBM's prediction layer: logistic regression (with and without L1 regularization), logistic regression with second-degree polynomial interactions, and a decision tree classifier. For fair comparison, all models are evaluated on the test set in its original imbalanced class distribution using F1-score, Cohen's κ, and AUPRC (see Table 1). We further motivate these metrics in Appendix D.

Using the refined concept set substantially improves results compared to the set of automatically generated concepts: A logistic regression model without regularization achieves 0.9198 F1, 0.8655 κ, and 0.9675 AUPRC, outperforming the in-production black-box classifier (0.9081 F1) and approaching GPT-4's performance (0.9485 F1). These findings demonstrate that expert-refined TBMs can perform highly competitively while maintaining interpretability and transparency.

Modeling concept interactions through polynomial features or tree-based predictors does not yield further gains, suggesting that concept-level dependencies are largely additive and well captured by a linear model. Inspection of the learned weights reveals that *Provider Switch* (-2.85), *Service Modification Intent* (-1.77), and *Relocation Mention* (-1.72) are the strongest churn indicators, while concepts such as *Technical Support Intent* (-0.33) and *Service Usage* (-0.24) show weaker or more ambiguous associations (see Table 12, Appendix J).

Model Efficiency. Substituting `gpt-4-turbo-128k` with `gpt-4o-mini` leads to a slightly worse predictive performance (2 percentage points in F1-score; Table 13) but reduces average inference time by more than half (1.03 ± 0.31s vs. 2.45 ± 0.54s per concept). This trade-off suggests that smaller models may be preferable for large-scale or latency-sensitive deployment scenarios, especially when integrated into interactive analytics workflows.

Overall, these results show that while fully automated concept generation remains challenging,

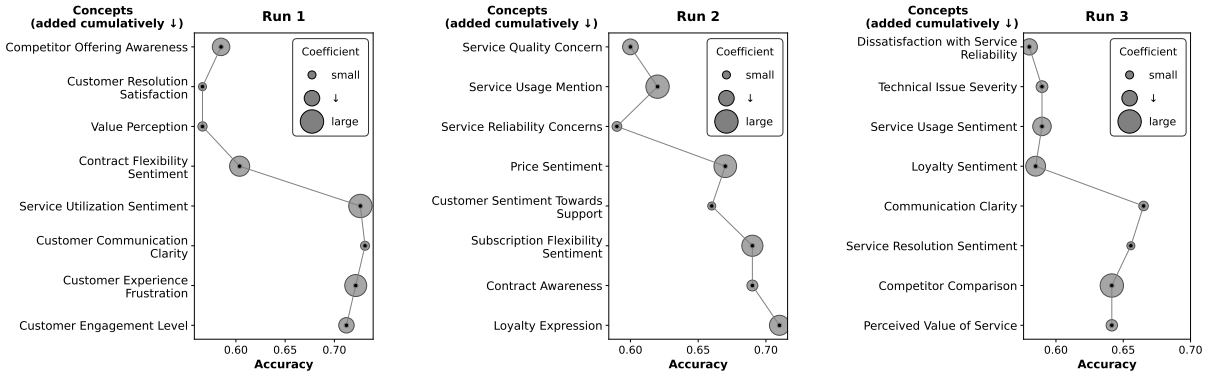

Figure 3: Learning curves for three independent, automated discovery runs. The y-axis lists concepts in their order of discovery (top-to-bottom); the x-axis represents the aggregate test accuracy achieved by the prediction layer using all concepts discovered up to that point. Node size indicates the absolute weight of each concept in the final classifier. The divergence in trajectories highlights the instability of automated concept generation.

Framework	Prediction Model	F1-Score	Cohen's κ	AUPRC
Black-Box Production Model	`BERTje`	0.9081	0.8429	–
Black-Box GPT-4	`gpt-4-turbo-128k`	0.9485	0.8996	–
TBM	Logistic Regression (no reg.)	0.9198	0.8655	0.9254
	Logistic Regression (L1-reg.)	0.9184	0.8621	0.9230
	Logistic + Interactions (no reg.)	0.8780	0.7984	0.8812
	Logistic + Interactions (L1-reg.)	0.8615	0.7752	0.8645
	Decision Tree	0.8835	0.8091	0.8522

Table 1: Performance comparison between black-box models and the TBM with `gpt-4-turbo-128k` as the concept measurement model and human-refined concepts. The interpretable-by-design TBM performs competitively with black-box models.

combining LLM-assisted discovery with expert refinement yields interpretable models that are competitive with high-performing black-box systems, making TBMs a practical and business-relevant alternative for explainable customer analytics.

5.2 Error analysis

To better understand the differences between the TBM and the black-box models, we manually analyze test samples where the best-performing configurations disagree.

The `BERTje`-based production model, limited to the first 512 tokens per call, often misclassifies cases where the churn intent appears later in the conversation. The GPT-4-based black-box model, despite processing full transcripts, also struggles when customers cancel or downgrade only part of their subscription, which is labeled as Churn but easily mistaken for No Churn.

In contrast, TBM errors are strongly tied to concept activations. Misclassifications typically occur when only weakly weighted concepts (e.g., *Service Usage* = "stopped using service") are triggered, providing insufficient evidence for churn, or when dissatisfaction-related concepts (e.g., *Price Sentiment*, *Subscription Flexibility Concern*) co-occur without an explicit switch intent. TBM also fails due to a lack of concept activations when customers confirm cancellation without any elaboration, whereas black-box models often succeed in those cases.

These findings highlight both the strengths and limits of concept-based reasoning. Refining definitions to distinguish partial from full churn, adjusting prompts to better capture intent, and incorporating complementary non-textual features (e.g., call frequency, duration, or tenure) could further improve robustness without compromising interpretability.

5.3 Identifying Call Profiles

Setup. To illustrate possible insights enabled by the TBM, we apply k-means clustering to the concept representations $\mathbf{s}(t)$ from both the train and the test set (400 samples in total). This allows us to group calls with similar semantic profiles and analyze their associated churn rates. We set $k = 4$, based on a qualitative elbow-method inspection.

Results. Figure 4 exemplifies the first two distinct cluster patterns and their associated churn rates. All four radar plots are provided with more details in Figure 6 (Appendix K). Cluster 1 is near-neutral on most concepts but shows high *Technical Support Intent* and *Service Usage*, suggesting engaged customers seeking help. Churn rates in this cluster are negligible, aligning with the assumption that these customers are still engaged and attempting to resolve problems. Cluster 2 scores low on both *Service Modification Intent* and *Service Usage*, pointing to disengaged customers, with a churn rate of 98%, suggesting that these are "already lost" customers. Cluster 3 exhibits negative *Price Sentiment*, frequent references to *Relocation*, signs of a *Provider Switch*, and discontinued *Service Usage*, reflecting customers who express dissatisfaction along with clear reasons for leaving. Their churn rate remains high at 80%. Cluster 4 shows moderate dissatisfaction, with negative *Price Sentiment* but positive *Technical Support Intent*, and a lower churn rate of 16%.

These findings can offer guidance for business decisions, e.g., by prioritizing retention efforts for engaged but dissatisfied customers (Cluster 4) while deprioritizing outreach to disengaged customers with likely churn (Cluster 2).

5.4 Interactive Dashboard

To support practical adoption, we developed an interactive dashboard deployed on the company's internal infrastructure. Users can retrieve a transcript by its identifier, trigger concept measurement, and obtain a churn prediction from the best-performing TBM model (Section 5.1). The interface displays concept activations, LLM reasoning, highlighted snippet evidence, and full concept descriptions. This makes the TBM's decision process transparent and accessible to non-technical stakeholders. Early internal feedback was highly positive, emphasizing the app's clarity and ease of use. An illustration of the interface is provided in Appendix L.

6 Discussion & Conclusion

Our results show that while automatically generated concepts in the TBM exhibit the typical trade-off between interpretability and predictive performance (Rudin, 2019; Gilpin et al., 2018), this gap largely disappears after domain-expert refinement. The refined TBM performs on par with black-box baselines, demonstrating that well-defined concepts can capture task complexity without sacrificing predictive performance.

Beyond performance, the TBM delivers clear business value by making model reasoning explicit and actionable. Concept activations help stakeholders understand why customers churn, group similar cases, and run "what-if" scenarios by adjusting concept scores and observing prediction changes. Its interpretable structure also enables human-in-the-loop refinement, where experts update concepts through prompt feedback to keep the model aligned with evolving business needs. This accessibility benefits both technical and non-technical users and has generated strong interest, with active steps now being taken towards deployment. Crucially, beyond churn, the TBM framework generalizes to any regression or classification task with definable intermediate concepts, making it a versatile tool for interpretable, business-aligned models across diverse operational workflows.

In sum, the TBM combines competitive performance with transparency and actionability. By translating complex language data into business-relevant concepts, it empowers decision-makers to understand, trust, and act on model insights, illustrating how interpretable NLP can drive measurable impact without compromising predictive performance in real-world customer retention.

7 Limitations and Future Work

The TBM offers both global and local interpretability: It identifies key concepts that drive predictions across the dataset, and for each instance, it reveals how strongly each concept is activated along with supporting snippets. However, one might argue that the framework merely shifts the black-box component from the prediction layer (as in the black-box models) to the Generation and Measurement Modules, which still rely on LLMs. While we assess the measurement step by evaluating robustness (Appendix G) and snippet sufficiency (Appendix I), these analyses do not establish whether the concept measurements align with human judgment.

Similarly, although the TBM's concept generation pipeline produces coherent and relevant concepts, we have not systematically evaluated them along dimensions such as clarity, redundancy, or business value. Due to limited resources, we were unable to conduct a thorough human evaluation or use a larger (human-annotated) dataset. The latter limits the strength of the conclusions we can draw

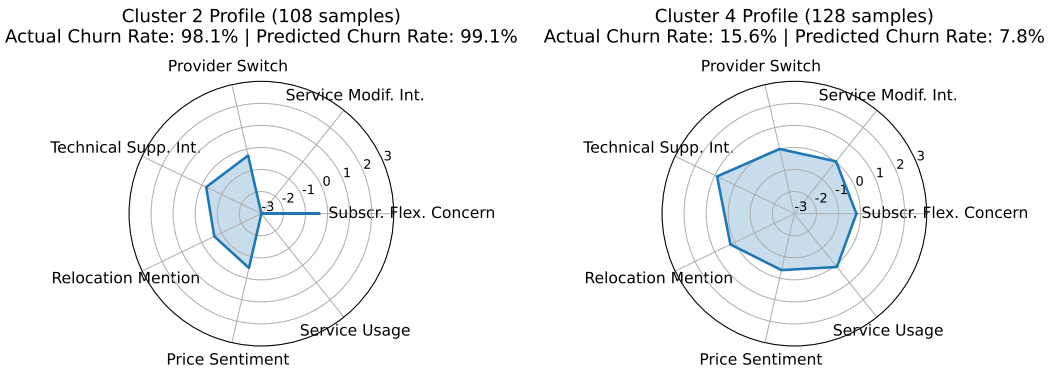

Figure 4: Results of k-means on concept representations. We show two out of four clusters with different characteristics and varying churn rates. The radar plots show the mean values of the concept activations within each cluster.

in terms of the robustness and generalization of the TBM. To address the former, we encourage future work to perform human-centered assessments of both the generation and measurement modules. For instance, Ludan et al. (2024) conducted expert evaluations of concept quality, using criteria such as relevance, redundancy, and difficulty, and human annotation of concept measurements. Applying such methodologies to our churn setting could help validate the TBM's interpretability and ensure that its outputs support actionable, trustworthy decisions.

Another limitation of the current implementation of the TBM-framework is its high computational cost. Unlike the black-box models, which require only a single prompt per input, the TBM involves a separate prompt for each concept, making it less feasible in settings with budget constraints or API rate limits. Although Ludan et al. (2024) propose batching multiple concepts into a single prompt to mitigate this issue, we found that this approach often compromised measurement quality. We suspect that this is likely due to the significantly longer and more complex nature of our input texts (e.g., multi-turn customer service dialogues) compared to the shorter texts used in datasets like CEBaB (Abraham et al., 2022) used by Ludan et al. (2024). Nonetheless, as LLM capabilities are expected to continue to improve, revisiting multi-concept prompting remains a promising direction. Alternatively, future work could explore the implementation of the approach by Sun et al. (2025), who replace the LLM-based concept measurement in their CB-LLM model with a sentence-embedding model (e.g., `all-mpnet-base-v2` from Huggingface (Wolf et al., 2020)), to estimate concept relevance through similarity between sentence embeddings of the input and concept descriptions. Another option, especially as the number of concepts grows, is to pre-process the transcript to identify relevant concepts, measure only those with the LLM, and assign the neutral response to the others.

Additionally, as our dataset consists of customer call transcripts, both the TBM framework and the black-box models we compare it with are constrained by errors introduced by the speech-to-text system used to generate these transcripts. Although internal procedures aim to ensure high transcription quality, we did not conduct a preliminary qualitative analysis of potential transcription errors.

To conclude, our experiments were limited to variants of logistic regression and decision trees, though many alternative white-box models exist (see Rudin (2019) for an overview). A particularly promising example is the globally interpretable additive model by Chen et al. (2018), which produces sparse, rule-based explanations that are well suited to our concept-level representation, especially for tasks where the TBM-framework involves a large number of concepts. We also only evaluated two LLMs of OpenAI's GPT-family. Future work should explore other LLMs, including open-sourced variants which could be finetuned to this domain. Finally, our `BERTje`-based classifier baseline was limited to the first 512 tokens per call. We encourage future work to include stronger baselines, such as a `BERT`-based classifier that processes each call transcript in chunks and aggregates the results.

Acknowledgements

VN is part of the project LESSEN with project number NWA.1389.20.183 of the research program NWA-ORC 2020/21 which is (partly) financed by the Dutch Research Council (NWO).

References

Eldar David Abraham, Karel D'Oosterlinck, Amir Feder, Yair Ori Gat, Atticus Geiger, Christopher Potts, Roi Reichart, and Zhengxuan Wu. 2022. CEBab: Estimating the causal effects of real-world concepts on NLP model behavior. In *Advances in Neural Information Processing Systems*.

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. 2018. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Jasmijn Bastings, Wilker Aziz, and Ivan Titov. 2019. Interpretable neural predictions with differentiable binary variables. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2963–2977, Florence, Italy. Association for Computational Linguistics.

Anna Bavaresco, Raffaella Bernardi, Leonardo Bertolazzi, Desmond Elliott, Raquel Fernández, Albert Gatt, Esam Ghaleb, Mario Giulianelli, Michael Hanna, Alexander Koller, Andre Martins, Philipp Mondorf, Vera Neplenbroek, Sandro Pezzelle, Barbara Plank, David Schlangen, Alessandro Suglia, Aditya K Surikuchi, Ece Takmaz, and Alberto Testoni. 2025. LLMs instead of human judges? a large scale empirical study across 20 NLP evaluation tasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 238–255, Vienna, Austria. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Chaofan Chen, Kangcheng Lin, Cynthia Rudin, Yaron Shaposhnik, Sijia Wang, and Tong Wang. 2018. An interpretable model with globally consistent explanations for credit risk. *Preprint*, arXiv:1811.12615.

Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. BERTje: A Dutch BERT Model. arXiv:1912.09582.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. 2018. Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 80–89.

Vincent Huang, Dami Choi, Daniel D. Johnson, Sarah Schwettmann, and Jacob Steinhardt. 2025. Predictive concept decoders: Training scalable end-to-end interpretability assistants. *Preprint*, arXiv:2512.15712.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, Online. Association for Computational Linguistics.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not Explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. 2020. Concept bottleneck models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5338–5348. PMLR.

Ryan Koo, Minhwa Lee, Vipul Raheja, Jong Inn Park, Zae Myung Kim, and Dongyeop Kang. 2024. Benchmarking cognitive biases in large language models as evaluators. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 517–545, Bangkok, Thailand. Association for Computational Linguistics.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117, Austin, Texas. Association for Computational Linguistics.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan,

Ce Zhang, Christian Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew A. Hudson, and 31 others. 2023. Holistic evaluation of language models. *Transactions on Machine Learning Research*. Featured Certification, Expert Certification, Outstanding Certification.

Josh Magnus Ludan, Qing Lyu, Yue Yang, Liam Dugan, Mark Yatskar, and Chris Callison-Burch. 2024. Interpretable-by-design text understanding with iteratively generated concept bottleneck. *Preprint*, arXiv:2310.19660.

Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA. Curran Associates Inc.

OpenAI. 2024. Gpt-4o model card.

Dheeraj Rajagopal, Vidhisha Balachandran, Eduard H Hovy, and Yulia Tsvetkov. 2021. SELFEXPLAIN: A self-explaining architecture for neural text classifiers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 836–850, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1135–1144, New York, NY, USA. Association for Computing Machinery.

Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215.

Emily Sheng and David Uthus. 2020. Investigating societal biases in a poetry composition system. In *Proceedings of the Second Workshop on Gender Bias in Natural Language Processing*, pages 93–106, Barcelona, Spain (Online). Association for Computational Linguistics.

Chung-En Sun, Tuomas Oikarinen, Berk Ustun, and Tsui-Wei Weng. 2025. Concept bottleneck large language models. In *The Thirteenth International Conference on Learning Representations*.

Zhen Tan, Lu Cheng, Song Wang, Bo Yuan, Jundong Li, and Huan Liu. 2024. Interpreting pretrained language models via concept bottlenecks. In *Advances in Knowledge Discovery and Data Mining*, pages 56–74, Singapore. Springer Nature Singapore.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2024. Explainability for large language models: A survey. *ACM Trans. Intell. Syst. Technol.*, 15(2).

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, LILI YU, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. LIMA: Less is more for alignment. In *Thirty-seventh Conference on Neural Information Processing Systems*.

A Methodology Details

Given a training set $\mathcal{D}_{\text{train}} = \{(t_i, y_i)\}_{i=1}^{N}$ of input texts t with labels y, and a test set $\mathcal{D}_{\text{test}}$, the Text Bottleneck Model (TBM) defines an interpretable intermediate representation based on K categorical concepts $\mathcal{C} = \{c_1, \ldots, c_K\}$.

Concept Measurement. For each concept $c_k \in \mathcal{C}$ and input text t, the measurement module produces a score

$$s(t, c_k) \in \mathbb{R},$$

where the value reflects the polarity and intensity of the concept in t (positive = supportive evidence, negative = adverse evidence, zero = absence or uncertainty).

The full concept representation of t is given by:

$$\mathbf{s}(t) = [s(t, c_1), s(t, c_2), \ldots, s(t, c_K)] \in \mathbb{R}^K.$$

Prediction Layer. A white-box classifier $f : \mathbb{R}^K \to \mathcal{Y}$ (e.g., logistic regression or decision tree) is trained on concept vectors:

$$\hat{y} = f(\mathbf{s}(t)).$$

Interpretability. The weights of f indicate the relative importance of each concept, giving *global explanations* across the dataset. On a sample-level, *local explanations* are obtained by the scores $s(t, c_k)$ and the supporting evidence snippets.

B LLM Selection

In our experiments, we use OpenAI's GPT-4 family of models accessed via the Azure OpenAI API. For concept generation and measurement, we rely on `gpt-4-turbo-128k`, one of the most capable publicly available models at the time. To assess performance trade-offs, we also evaluate the smaller and cheaper `gpt-4o-mini` for concept measurement. Both GPT models support a 128k-token context window, enabling them to handle full-call transcripts with multiple conversational turns. We note that this work focuses on thoroughly evaluating the proposed approach and its applicability to the churn classification task, rather than comparing different LLMs for the TBM. Exploring a broader set of models, including open-source alternatives such as Mistral (Jiang et al., 2023), is left for future work.

C Hyperparameter Tuning

We tune hyperparameters via 5-fold cross-validation on $\mathcal{D}_{\text{train}}$: inverse regularization strength $C \in \{0.01, 0.1, 1, 10, 100\}$ for L1-regularized logistic regression, and `max_depth` $\in \{3, 5, 10, 15\}$, `min_samples_split` $\in \{2, 5, 10\}$, and `min_samples_leaf` $\in \{1, 2, 4\}$ for decision trees.

D Metrics

Due to the class imbalance in churn data, we use F1-score, Cohen's κ and AUPRC to provide a more rigorous evaluation than standard accuracy. The F1-score provides a balanced harmonic mean of precision and recall, ensuring the model effectively captures churning customers without excessive false positives. To ensure these results are not inflated by the class distribution, Cohen's κ to measure the agreement between predictions and reality beyond what would be expected by random chance. Finally, the Area Under the Precision-Recall Curve (AUPRC) evaluates the model's performance across all classification thresholds and is a function of the model's confidence in its response, offering a more sensitive assessment of minority class identification than standard accuracy or ROC curves.

E Concept Measurement Example

We report a sample concept measurement of the concept *Provider Switch Mentioned* on a dummy transcript in Figure 5. [company] denotes a redacted competitor name and was applied manually after model inference.

F Refined Concept Details

F.1 Obtaining Refined Concepts

TelCo's domain experts. The domain expert group at TelCo that we collaborated with in this work is an internal cross-functional team focusing on churn prevention. For this project, the group provided business input to ensure that the extracted concepts aligned with actionable business levers (e.g., pricing, service modification, contract flexibility).

Refinement process. An initial set of promising concepts was written manually by the authors and refined using a zero shot prompting of `gpt-4-turbo-128k`. This resulted in the concepts *Price Sentiment* and *Provider Switch*. Through automatic generation, the concepts *Service Usage*, *Technical Support Intent*, *Subscription Flexibility Concern*, and *Communication Style* were identified. These were then manually reviewed and refined by the authors and then further revised in collaboration with the churn analysts group. Refinement primarily involved: simplifying wording for clarity, enriching concept descriptions with examples, remove overlapping options from the response guide enriching the response guide to clearly separate the different options, ensuring actionability from a business perspective. Given that our expert-refined concept set is non-exhaustive, we anticipate that performance can be further enhanced by incorporating additional concepts to capture more niche churn signals.

Changes made. From the initial pool of concepts:

- **Kept:** *Price Sentiment*, *Provider Switch*, *Subscription Flexibility Concern*, *Service Usage*, *Technical Support Intent*
- **Modified:** All five concepts were refined by expanding the concept description and adding examples in the response guide.
- **Added:** Two new concepts were proposed by the domain experts to capture business-specific factors not suggested by the model: *Relocation Mention*, *Service Modification Intent*

- **Removed:** *Communication Style*, since it was not actionable and not directly related to Churn.

F.2 Concept Details

For transparency, we include the final definitions of the concepts used in the Text Bottleneck Model. To protect commercially sensitive details, some fields (response guide) have been redacted (explicitly stated), while other fields (concept description, concept question) have been simplified (in *italics*). We present the full version of the concept *Relocation Mention* to give an impression of the overall level of detail of the concept definitions.

Key	Value
Concept Name	Relocation Mention
Concept Description	Relocation Mention captures whether the customer mentions moving to a new home, and whether this move is associated with continuing or canceling [TelCo] service.
Concept Question	Does the customer mention moving houses, and do they express an intent to cancel or continue their [TelCo] service as a result?
Possible Responses	Moving and Canceling Service, Possible Relocation Mentioned, No Mention of Relocation, Moving and Continuing Service
Response Guide	**Moving and Canceling Service:** The customer clearly states that they are moving and plan to cancel their [TelCo] service or taking a subscription from a different provider at the new address. Examples: "I am moving, so I am cancelling everything.", "We are not getting any more [TelCo] in the new house.". **Possible Relocation Mentioned:** There is an indirect mention or hint of a move, without confirmation or clarity on service continuation or cancellation. Examples: "We are looking for a new house.", "Maybe we will move this year.". **No Mention of Relocation:** There is no mention of moving houses or any change in living situation during the call. **Moving and Continuing Service:** The customer mentions moving but also expresses an intent to keep using [TelCo], such as arranging a transfer of service. Examples: "We are moving soon, can I take [TelCo] with me?", "I want to connect internet at the new address.".
Response Mapping	Moving and Canceling Service: -3, Possible Relocation Mentioned: -1, No Mention of Relocation: 0, Moving and Continuing Service: +3

Table 2: JSON Representation for the concept *Relocation Mention*.

Key	Value
Concept Name	Price Sentiment
Concept Description	*Captures whether the customer expresses negative, neutral, or positive sentiment towards the price of [TelCo].*
Concept Question	*Does the customer express their opinion about the prices of [TelCo] during the call?*
Possible Responses	Satisfied, Neutral, Dissatisfied
Response Guide	*Redacted for commercial sensitivity.*
Response Mapping	Satisfied: +3, Neutral: 0, Dissatisfied: -3

Table 3: JSON Representation for the concept *Price Sentiment*.

Key	Value
Concept Name	Provider Switch
Concept Description	*Captures whether the customer refers to switching between providers (either to or from [TelCo]).*
Concept Question	*Does the customer mention a provider switch?*
Possible Responses	Competitor Mentioned Directly, Competitor Implied, No Provider Switch Mentioned, Switch to [TelCo]
Response Guide	*Redacted for commercial sensitivity.*
Response Mapping	Competitor Mentioned Directly: -3, Competitor Implied: -1, No Provider Switch Mentioned: 0, Switch to [TelCo]: +3

Table 4: JSON Representation for the concept *Provider Switch*.

Key	Value
Concept Name	Service Modification Intent
Concept Description	*Captures whether the customer shows intent to change their existing service package (e.g., upgrade or downgrade).*
Concept Question	*What is the customer's intention regarding modification of their service?*
Possible Responses	Downgrade Service, Inquire About Downgrading, No Clear Modification Intent, Inquire About Upgrading, Upgrade Service
Response Guide	*Redacted for commercial sensitivity.*
Response Mapping	Downgrade Service: -3, Inquire About Downgrading: -1, No Clear Modification Intent: 0, Inquire About Upgrading: +1, Upgrade Service: +3

Table 5: JSON Representation for the concept *Service Modification Intent*.

Sample Transcript:

```
[agent][00:00]: Good afternoon, welcome to [TelCo] customer service. How can I help you?

[customer][00:04]: Hello, I am moving in two weeks and do not want to renew my contract.
                   I just wanted to discuss what my options are.

[agent][00:10]: Of course. May I ask what the reason is that you do not want to continue the
                contract?

[customer][00:14]: I have actually been very satisfied with the price and service from [TelCo].
                   That is not the issue.

[agent][00:18]: Good to hear. What makes you want to stop then?

[customer][00:22]: When moving, I prefer some flexibility. I saw that [company] has a moving offer
                    where I get a shorter contract. With [TelCo], I am immediately tied down for another
                    year, and that just doesn't fit my situation right now. With other providers, you
                    can choose a shorter contract.

[agent][00:36]: I get it. So you are looking for something that fits better with a temporary living
                situation?

[customer][00:40]: Exactly. I don't know yet how long I will stay at the new address. I don't know yet
                   if I will stay there or have to move again afterwards, so a long-term contract feels
                   like a risk right now.

[agent][00:48]: Clear, thanks for the explanation. I would be happy to look into whether we can offer
                something more flexible.
```

Measured Response: `competitor mentioned directly`

Snippets Dictionary:

```
{
'competitor mentioned directly': [
'I saw that [company] has a relocation offer where I get a shorter contract',
'I am switching to [company]. Their offer fits better at the moment'
],
'competitor implied': [
'other providers aren't so difficult about that'
],
'no provider switch mentioned': [],
'switch to TelCo': []
}
```

Thoughts: `The customer explicitly mentions [company] as the competitor they are considering switching to, citing specific features of their offer that better meet their needs. There is also an implied mention of other competitors in general, but the focus is on [company]. There is no indication of switching to [TelCo].`

Figure 5: Illustrative conversation sample and annotation.

Key	**Value**
Concept Name	Service Usage
Concept Description	*Captures whether the customer indicates active, discontinued, or no mention of service usage.*
Concept Question	*Does the customer indicate using or having stopped using [TelCo] services?*
Possible Responses	Stopped Using Service, No Mention of Service Usage, Using Service
Response Guide	*Redacted for commercial sensitivity.*
Response Mapping	Stopped Using Service: -3, No Mention of Service Usage: 0, Using Service: +3

Table 6: JSON Representation for the concept *Service Usage*.

Key	Value
Concept Name	Subscription Flexibility Concern
Concept Description	*Captures whether the customer expresses satisfaction, dissatisfaction, or neutrality about the flexibility of subscription terms (e.g., contract duration, cancellation, modifications).*
Concept Question	*Does the customer comment on the flexibility of their subscription?*
Possible Responses	Dissatisfied with Flexibility, Neutral or No Mention of Flexibility, Satisfied with Flexibility
Response Guide	*Redacted for commercial sensitivity.*
Response Mapping	Dissatisfied with Flexibility: -3, Neutral or No Mention of Flexibility: 0, Satisfied with Flexibility: +3

Table 7: JSON Representation for the concept *Subscription Flexibility Concern*.

Key	Value
Concept Name	Technical Support Intent
Concept Description	*Captures whether the customer is seeking technical support, or expressing unresolved technical frustration.*
Concept Question	*Does the customer mention technical issues or ask for technical support?*
Possible Responses	Technical Issues and No Longer Seeking Help, No Mention of Technical Issues or Support, Seeking Help with Technical Issue
Response Guide	*Redacted for commercial sensitivity.*
Response Mapping	Technical Issues and No Longer Seeking Help: -3, No Mention of Technical Issues or Support: 0, Seeking Help with Technical Issue: +3

Table 8: JSON Representation for the concept *Technical Support Intent*.

G Concept Measurement Robustness

Setup. Given the length and occasional noisiness of call transcripts, we assess the robustness of the concept measurement process by repeating it three times on the combined dataset $\mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{test}}$. We evaluate consistency using triple agreement (the proportion of samples where all three runs yield the same measurement response) and Fleiss' Kappa κ (Fleiss, 1971), a standard inter-rater reliability metric.

Results. The results indicate high stability, with an average triple agreement of 0.943 ± 0.027 and a Fleiss' Kappa κ of 0.903 ± 0.027 across concepts, suggesting near-perfect consistency. Qualitative inspection further supports this, as disagreements were limited to adjacent response classes (e.g., switching between "downgrade service" and "inquire about downgrading"). Concept-specific results are provided in Table 9.

Concept	Triple Agreem.	Fleiss Kappa
Provider Switch	0.96	0.905
Price Sentiment	0.94	0.872
Relocation Mention	0.98	0.956
Tech. Supp. Intent	0.92	0.897
Serv. Mod. Intent	0.94	0.916
Subs. Flex. Concern	0.96	0.885
Service Usage	0.90	0.893
Average	**0.943 ± 0.027**	**0.903 ± 0.027**

Table 9: Agreement metrics for each concept. Triple Agreement shows the proportion of samples with identical labels across three runs. Fleiss Kappa measures inter-rater agreement accounting for chance. The average and standard deviation (±) are reported across concepts.

H Chain-of-thought ablation

Setup. Prior work has shown that prompting LLMs to explicitly reason about their answers can improve task performance, which is known as "Chain-of-Thought"-prompting (Wei et al., 2022). In the TBM framework, the measurement model is asked to include a brief "thoughts" section, explaining its final response based on the extracted snippets. To test whether this reasoning step contributes to performance, we re-run the experiment without requesting the thoughts section.

Results. Using `gpt-4-turbo-128k` and the best-performing prediction layer configuration (see Table 1), validation accuracy drops to 85.35%, F1-score to 84.49%, and AUROC to 91.97%. This finding suggests that prompting the model to reason about the extracted evidence helps it arrive at more accurate and robust concept measurements, which in turn lead to better prediction performance.

I Snippet Analysis and Sufficiency

Setup. The extracted snippets associated with each concept measurement are central to the local interpretability of the TBM framework (e.g., "I'm not using the TV anymore." is a representative snippet for "Stopped using service"). To analyze snippet quality, we first report concept-level statistics, such as the number of snippets per measurement and the proportion of exact, non-fuzzy matches in the original transcript. Next, we assess whether the extracted snippets alone provide sufficient evidence to accurately recover the original concept measurement. We conduct two experiments: (i) We provide `gpt-4-turbo-128k` with a concept and a dictionary where each response is paired with its extracted snippets, and ask which response is best supported. However, this can be trivial since the measurement model often extracts snippets only for the selected response, revealing the answer via a single non-empty entry. (ii) To address this, we repeat the experiment using only the snippets of the selected response, without any dictionary, to see if an independent LLM draws the same conclusion. For both experiments, we exclude cases with no extracted snippets, as these typically indicate neutral responses and offer no basis for evaluation, leaving 2043 of 2800 measurements.

Results. Table 10 summarizes key properties of the extracted snippets per concept. Notably, on average, 0.88 ± 0.01 of the snippets appear as direct matches in the original transcript. Through qualitative analysis, we observe that deviations typically occur due to preprocessing artifacts (e.g., ellipses added during anonymization) or transcription noise, which the model sometimes subtly "repairs" to restore fluency. As for snippet sufficiency, we find that an independent LLM can accurately predict the correct concept measurement based on the snippets: $90.6 \pm 3.2\%$ accuracy using the full snippet dictionary, and $87.1 \pm 1.9\%$ using only the selected response's snippets. These results suggest that the TBM framework produces snippet evidence that is both faithful to the input text and sufficient to support its concept-level decisions. Preliminary qualitative analysis supports this sufficiency. Full per-concept results are provided in Table 11.

Concept	Total Snippets	Avg. Length	Avg. Snippets / Transcript	Prop. Found	Prop. 0 Resp.
Provider Switch	608	72.45 ± 41.06	1.48 ± 1.30	0.88	0.33
Price Sentiment	205	84.53 ± 54.06	0.50 ± 0.93	0.88	0.72
Relocation Mention	528	72.31 ± 40.49	1.29 ± 1.23	0.86	0.39
Service Modification Intent	855	73.30 ± 41.96	2.09 ± 1.13	0.90	0.11
Service Usage	831	64.96 ± 37.31	2.03 ± 1.25	0.88	0.12
Subscription Flexibility Concern	629	84.51 ± 52.26	1.53 ± 1.38	0.89	0.33
Technical Support Intent	1147	86.17 ± 49.87	2.80 ± 1.16	0.89	0.01
Average	**686.14**	$\mathbf{76.66 \pm 45.64}$	$\mathbf{1.67 \pm 1.38}$	$\mathbf{0.88 \pm 0.01}$	$\mathbf{0.29 \pm 0.22}$

Table 10: Concept-level snippet statistics based on 400 transcripts $\times$ 7 concepts = 2800 concept measurements: total snippet count, average snippet length in characters (mean $\pm$ std), average number of snippets per transcript (mean $\pm$ std), the proportion of snippets found as a perfect, non-fuzzy match in the original transcript, and the proportion of samples where no snippets were extracted for any response. The average and standard deviation (±) are reported across concepts.

Concept	Full Dict (All)	Meas. Resp. Snippets (All)	Full Dict (Non-empty)	Meas. Resp. Snippets (Non-empty)
Provider Switch	0.993	0.922	0.993	0.886
Price Sentiment	0.988	0.971	0.957	0.896
Relocation Mention	0.988	0.917	0.980	0.864
Service Mod. Intent	0.997	0.868	0.997	0.851
Service Usage	0.971	0.876	0.967	0.859
Subscription Flex. Concern	0.968	0.898	0.952	0.846
Technical Support Intent	0.988	0.893	0.988	0.892
Average	**0.985 ± 0.010**	**0.906 ± 0.032**	**0.976 ± 0.016**	**0.870 ± 0.019**

Table 11: Concept label prediction accuracy using full dictionaries vs. extracted snippets of the measured response. 'All' includes all 2870 samples, 'Non-Empty' includes only samples with at least one extracted snippet. The average and standard deviation (±) are reported across concepts.

J Prediction Performance

Concept	Coefficient
Provider Switch	-2.85
Service Modification Intent	-1.77
Relocation Mention	-1.72
Price Sentiment	-0.67
Subscription Flexibility Concern	-0.61
Technical Support Intent	-0.33
Service Usage	-0.24

Table 12: Concept importance based on logistic regression coefficients, sorted from most to least negative. More negative values indicate stronger association with the "Churn" label.

Framework (Measurement Model)	Prediction Model	F1-Score	Cohen's κ	AUPRC
Black-Box Production Model (n.a.)	`BERTje`	0.9081	0.8429	–
GPT-4 Black-Box (n.a.)	`gpt-4-turbo-128k`	**0.9485**	**0.8996**	–
	`gpt-4o-mini`	*0.9307*	*0.8752*	–
TBM (`gpt-4-turbo-128k`)	Logistic Regression (no reg.)	<u>0.9198</u>	<u>0.8655</u>	**0.9254**
	Logistic Regression (L1-reg.)	0.9184	0.8621	0.9230
	Logistic + Interactions (no reg.)	0.8780	0.7984	0.8812
	Logistic + Interactions (L1-reg.)	0.8615	0.7752	0.8645
	Decision Tree	0.8835	0.8091	0.8522
TBM (`gpt-4o-mini`)	Logistic Regression (no reg.)	0.9026	0.8315	<u>0.9098</u>
	Logistic Regression (L1-reg.)	0.8889	0.8122	0.8955
	Logistic + Interactions (no reg.)	0.8571	0.7680	0.8504
	Logistic + Interactions (L1-reg.)	0.8358	0.7410	0.8291
	Decision Tree	0.8744	0.7955	0.8388

Table 13: Performance comparison between two black-box models (`BERTje`- and GPT-4-based black-box model) and the TBM with refined concepts. The best-performing model is highlighted in **bold**, second-best in *italics*, third-best is <u>underlined</u>.

K Identifying Call Profiles

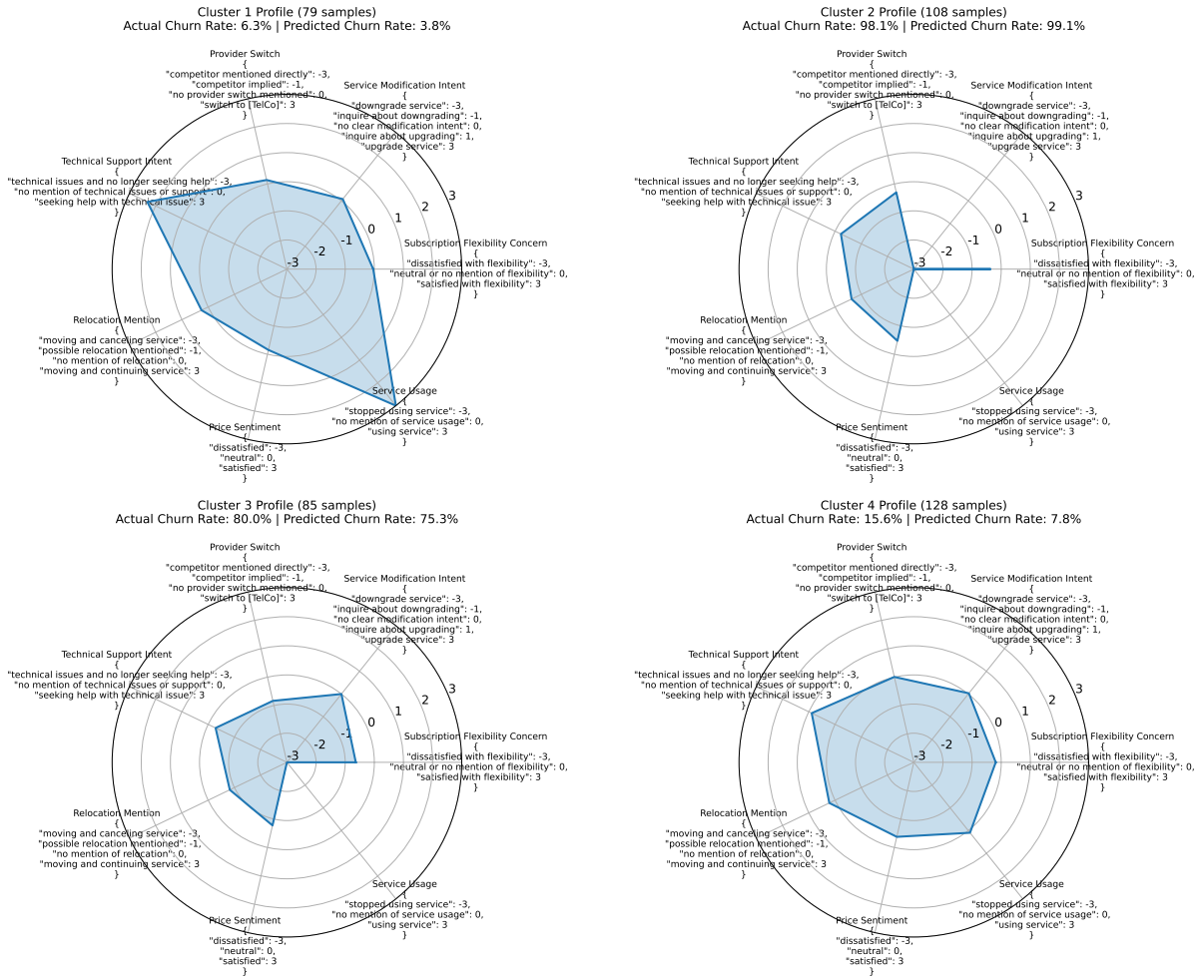

Figure 6: Results of k-means on concept representations. We identify four distinct clusters with different characteristics and varying churn rates. The radar plots show the mean values of the concept responses within each cluster.

L Interactive dashboard

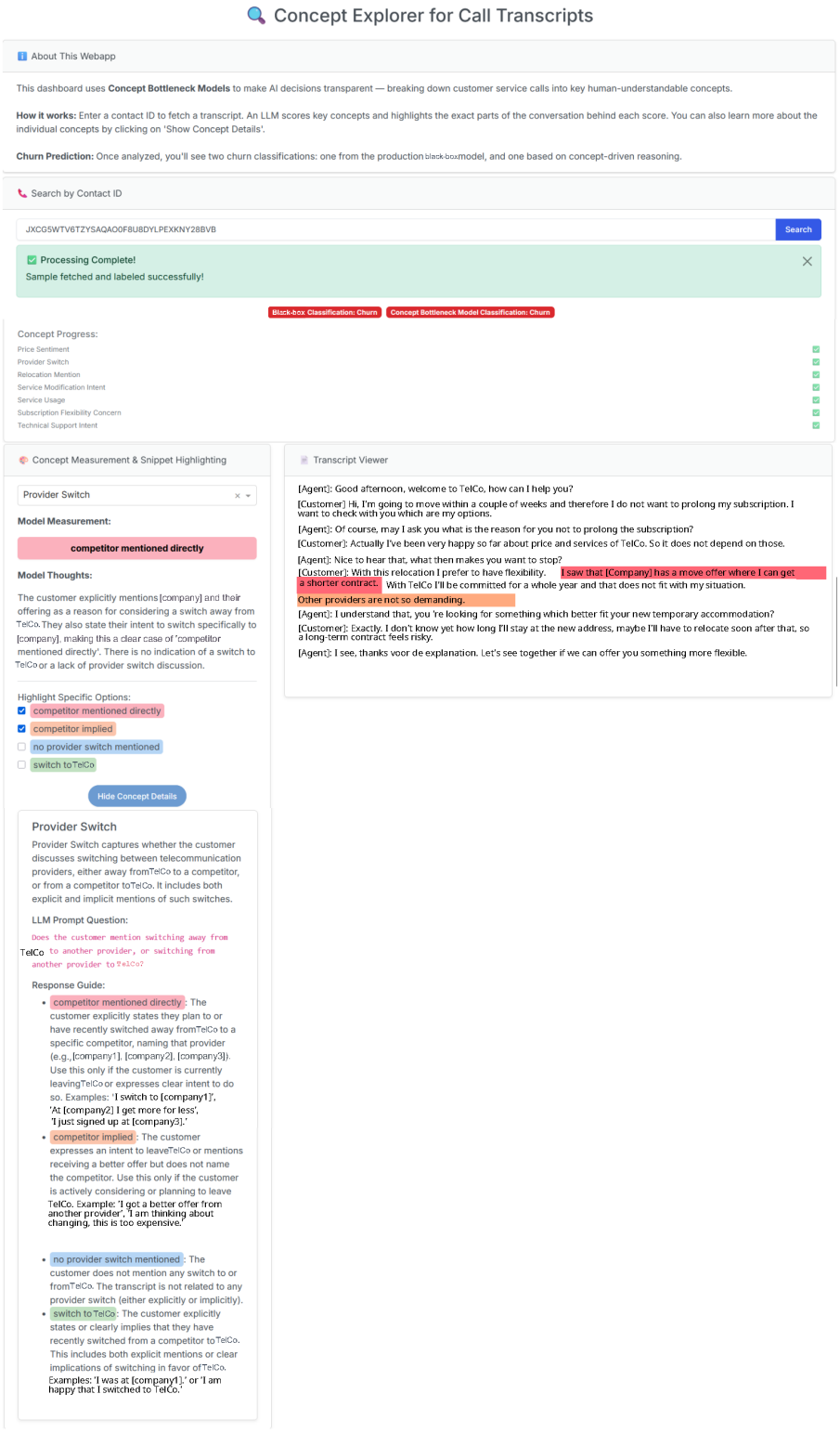

Figure 7: Screenshot of the interactive dashboard on a manually created transcript. Users can enter the unique contact ID of a call transcript they want to analyze with the TBM. Upon completion, they see the predicted label of the black-box production model and TBM. Then, they can explore the individual concept measurements and automatically highlight the extracted snippets in the transcript, colored by response option. They can also explore the model thoughts and the concept details. [company] refers to a redacted competitor and was applied manually after model inference.

M Prompts

```
Concept Feature Engineering Task (Redacted)

You are an expert data scientist working for [TelCo], a large European telecommunications company. The company wants to predict
whether customers will churn and, more importantly, identify the key business-relevant reasons for churn. This analysis is
based on customer call transcripts labeled with binary categories:
- "Churn" (customer contacts [TelCo] with the intention of terminating their subscription)
- "No Churn" (customer contacts [TelCo] for another reason and continues their subscription)

Your task is to propose a new concept that helps distinguish between the labels.
A description of what a good concept contains was provided internally, but is not included here for confidentiality reasons.

Concept Definition Format (simplified):
{
 "Concept Name": "...",
 "Concept Description": "...",
 "Concept Question": "...",
 "Possible Responses": [...],
 "Response Guide": "...",
 "Response Mapping": {...}
}

Examples of accepted and rejected concepts, as well as representative transcripts,
were provided internally to guide generation but are not included here for
confidentiality reasons.

Definition:
```

Figure 8: Concept Generation Prompt. Words in curly brackets (e.g., {concept}) indicate placeholders and are dynamically replaced with specific values depending on the context in which the prompt is applied. [company] denotes a redacted competitor name and was applied manually after model inference.

```
Concept Improvement Task

You are an expert data scientist working for [TelCo], a leading European telecommunications company. [TelCo] wants to predict customer
churn while also identifying the key business-relevant reasons that drive churn. This analysis is based on customer
service call transcripts
labeled as follows:
- "Churn": The customer contacts [TelCo] with the intent of terminating their subscription.
- "No Churn": The customer contacts [TelCo] for another reason and continues their subscription.

You are given a concept that might need improvement.
A description of potential areas of improvements was provided internally (e.g., validity, clarity, formatting),
but is not included here for confidentiality reasons.

Your task is to return information about any potential problems in the concept along with the improved concept.
If the concept is already well-formed and requires no changes, return the original concept with `"None"` for other responses.

Examples of flawed concepts and their improved versions were provided internally,
but are not included here for confidentiality reasons.

---
Below is the concept for you to improve.
{concept}
{user_feedback (optional additional feedback from the user)}
Response:
```

Figure 9: Concept Improvement Prompt. Words in curly brackets (e.g., {concept}) indicate placeholders and are dynamically replaced with specific values depending on the context in which the prompt is applied. [company] denotes a redacted competitor name and was applied manually after model inference.

```
Concept Measurement Task

You work for [TelCo], a European telecom provider. Your task is to analyze a customer call transcript and determine whether
a certain concept appears in the conversation.

You will receive:
- A concept definition
- A customer call transcript

Your job is to:
1. Read the entire transcript carefully from beginning to end. Important clues may appear late in the conversation.
2. Extract short, specific snippets that directly support each possible label in the response guide.
3. Reason about the classification using only the snippets.
4. Select the best-fitting classification — the one most directly supported by the content.

A set of guidelines was provided internally (e.g., "use only what is explicitly stated in the transcript",
"follow the exact response format"), but is not shown here for confidentiality reasons.

An example of how the task is expected to be performed was provided internally,
but is not shown here for confidentiality reasons.

Now, it is your turn to complete the task for the concept below.
Do not infer — classify based only on clear, direct statements.
Check the whole transcript before answering.

---
Concept:
{concept}

Transcript: {transcript}
Response JSON:
```

Figure 10: Concept Measurement Prompt. Words in curly brackets (e.g., {concept}) indicate placeholders and are dynamically replaced with specific values depending on the context in which the prompt is applied. [company] denotes a redacted competitor name and was applied manually after model inference.

```
Churn Classification Task

You work for [TelCo], a European telecom provider. Your task is to analyze a customer call transcript and determine
whether the main purpose of the call is about churning — meaning the customer
wants to cancel all or part of their [TelCo] subscription.

You will receive:
- A full transcript of a customer service call.

Your task is to:
1. Read the entire transcript carefully. Important clues may appear at any point.
2. Extract short, specific snippets that directly support either 'Churn' or 'No Churn'.
3. Based on this evidence, decide which label fits best.
4. Return the result using the exact JSON format below.

Guidelines:
A set of clean guidelines was provided internally (e.g., "use only what is explicitly stated in the transcript",
"follow the exact response format"), but is not shown here for confidentiality reasons.

An example of how the task is expected to be performed was provided internally, but is not shown here for confidentiality reasons.

---
Now analyze the following transcript. Return only the JSON response. Do not add any other text.

Transcript:
{transcript}
Response JSON:
```

Figure 11: Black-Box Model Prompt. Words in curly brackets (e.g., {transcript}) indicate placeholders and are dynamically replaced with specific values depending on the context in which the prompt is applied. [company] denotes a redacted competitor name and was applied manually after model inference.