IRSTI 28.23.37; 81.93.29 DOI: https://doi.org/10.26577/JMMCS201211426

A. Turarbek[1], S. Narbayeva[1*], A. Dokuz[2], M. Karimov[1], L. Kopbossyn[1]
[1]Farabi University, Almaty, Kazakhstan
[2]Niğde Ömer Halisdemir University, Nigde, Turkey
*e-mail: Narbaeva.salta@kaznu.kz

A BLOCKCHAIN-ENHANCED RETRIEVAL-AUGMENTED GENERATION FRAMEWORK FOR DISINFORMATION DETECTION

Automated assessment of claim veracity remains an open problem in natural language processing, particularly for political statements whose truthfulness is ambiguous, contextual and subject to expert disagreement. This paper presents VeraChain, a hybrid system that combines a fine-tuned DistilBERT classifier, a dual-index retrieval stage (retrieval-augmented classification in the sense of RAG; no text is generated), and a hash-chained ledger that records every verdict together with its evidence and, for borderline predictions, supplies the verdicts of similar past statements as a consensus signal. Both FAISS indices and the ledger are built from the LIAR training split only, and the ledger is frozen for the main test evaluation. A stepwise ablation over five random seeds on the 1,283 LIAR test statements shows that DistilBERT alone reaches 64.36 ± 1.25 percent accuracy; adding a single index of REAL statements lowers accuracy by 0.75 percentage points, adding a second index of FAKE statements recovers this loss (+0.87 points, positive in four of five seeds), whereas the probability-ratio score and the ledger consensus step add no consistent gain (-0.05 and -0.11 points). With the original seed, VeraChain reaches 65.24 percent against 64.77 percent for the same system without the consensus step (exact McNemar $p = 0.36$; 95 percent bootstrap interval of the difference $[-0.39, 1.33]$ points). Retrieval fusion lowers the expected calibration error of DistilBERT from 0.112 to 0.045, although temperature scaling of DistilBERT alone gives 0.031. On a single laptop GPU, the median processing time is 17.5 milliseconds per statement (95th percentile 20.9 ms), including the write of the ledger entry to disk. The results show that a hash-chained ledger can provide tamper-evident audit records for automated veracity assessment at a small computational cost, whereas its use as a consensus signal does not improve accuracy on LIAR.

Keywords: claim veracity assessment, disinformation, blockchain, retrieval-augmented classification, DistilBERT, FAISS, LIAR dataset, ablation study, audit trail, transformer models.

А.Т. Турарбек[1], С.М. Нарбаева[1*], А.Ш. Докуз[2], М.С. Каримов[1], Л.С. Көпбосын[1]
[1]Әл-Фараби атындағы Қазақ ұлттық университеті, Алматы, Қазақстан
[2]Өмер Халисдемир атындағы Нигде университеті, Нигде, Түркия
*e-mail: Narbaeva.salta@kaznu.kz

Жалған ақпаратты анықтауға арналған блокчейнмен күшейтілген іздеу арқылы толықтырылған генерация жүйесі

Саяси мәлімдемелердің ақиқаттығын автоматты түрде бағалау табиғи тілді өңдеудегі шешілмеген мәселе болып қала береді. Мақалада дәл бапталған DistilBERT жіктеуішін, қос индексті іздеу кезеңін (RAG мағынасындағы іздеумен толықтырылған жіктеу; мәтін генерацияланбайды) және әрбір шешімді оның дәлелдерімен бірге жазатын, ал шекаралық болжамдар үшін ұқсас бұрынғы мәлімдемелердің шешімдерін консенсус сигналы ретінде беретін хэш-тізбекті тізілімді біріктіретін VeraChain жүйесі ұсынылады. Екі FAISS индексі де, тізілім де тек LIAR оқыту жиынынан құрылады, ал негізгі тест кезінде тізілім өзгертілмейді (қатырылады). LIAR-дың 1283 тест мәлімдемесінде бес кездейсоқ seed бойынша жүргізілген кезеңдік абляция DistilBERT-тің өзі $64{,}36 \pm 1{,}25\%$ дәлдікке жететінін көрсетті; тек шынайы мәлімдемелердің бір индексін қосу дәлдікті 0,75 пайыздық тармаққа төмендетеді, жалған мәлімдемелердің екінші индексі бұл шығынды қалпына келтіреді (+0,87 п.т., бес seed-тің төртеуінде оң), ал ықтималдықтар қатынасы мен тізілім консенсусы тұрақты пайда бермей-

© 2026 Al-Farabi Kazakh National University

ді (−0,05 және −0,11 п.т.). Бастапқы seed-пен VeraChain 65,24% дәлдікке жетеді, консенсус қадамы жоқ дәл сол жүйе үшін бұл 64,77% (McNemar $p = 0{,}36$; айырманың 95% бутстреп аралығы [−0,39; 1,33] п.т.). Іздеу нәтижесін біріктіру DistilBERT-тің калибрлеу қатесін (ECE) 0,112-ден 0,045-ке дейін азайтады, бірақ DistilBERT-ті температуралық масштабтау 0,031 береді. Бір ноутбук GPU-ында бір мәлімдемені өңдеудің медианалық уақыты тізілім жазбасын дискіге жазуды қоса алғанда 17,5 мс (95-процентиль 20,9 мс). Нәтижелер хэш-тізбекті тізілім аз есептеу шығынымен өзгертуден қорғалған аудит жазбаларын бере алатынын, бірақ оны консенсус сигналы ретінде пайдалану LIAR-да дәлдікті арттырмайтынын көрсетеді.
Түйін сөздер: мәлімдеменің ақиқаттығын бағалау, жалған ақпарат, блокчейн, іздеумен толықтырылған жіктеу, DistilBERT, FAISS, LIAR деректер жинағы, абляциялық талдау, аудит ізі, трансформер модельдері.

А.Т. Турарбек[1], С.М. Нарбаева[1*], А.Ш. Докуз[2], М.С. Каримов[1], Л.С. Копбосын[1]
[1]Казахский национальный университет имени аль-Фараби, Алматы, Казахстан
[2]Университет Нигде имени Омера Халисдемира, Нигде, Турция
*e-mail: Narbaeva.salta@kaznu.kz

Система дополненной поиском генерации с использованием блокчейна для обнаружения дезинформации

Автоматическая оценка достоверности утверждений остаётся нерешённой задачей обработки естественного языка, особенно для политических заявлений, истинность которых неоднозначна, зависит от контекста и вызывает разногласия экспертов. В статье представлена VeraChain — гибридная система, объединяющая дообученный классификатор DistilBERT, этап поиска по двум индексам (классификация, дополненная поиском, в смысле RAG; текст не генерируется) и хэш-цепочечный реестр, который записывает каждый вердикт вместе с доказательствами, а для пограничных предсказаний предоставляет вердикты похожих прошлых утверждений в качестве сигнала консенсуса. Оба индекса FAISS и реестр строятся только по обучающей части LIAR, и для основного теста реестр замораживается. Поэтапная абляция по пяти случайным seed на 1283 тестовых утверждениях LIAR показывает, что DistilBERT сам по себе достигает точности $64{,}36 \pm 1{,}25\%$; добавление одного индекса истинных утверждений снижает точность на 0,75 процентного пункта, второй индекс ложных утверждений возвращает эту потерю (+0,87 п.п., положительно в четырёх из пяти seed), а отношение вероятностей и консенсус по реестру устойчивого выигрыша не дают (−0,05 и −0,11 п.п.). С исходным seed VeraChain достигает 65,24% против 64,77% у той же системы без шага консенсуса (точный тест Макнемара $p = 0{,}36$; 95%-й бутстреп-интервал разности [−0,39; 1,33] п.п.). Объединение с поиском снижает ожидаемую ошибку калибровки DistilBERT с 0,112 до 0,045, однако температурное масштабирование одного DistilBERT даёт 0,031. На одном GPU ноутбука медианное время обработки утверждения составляет 17,5 мс (95-й процентиль 20,9 мс) с учётом записи блока реестра на диск. Результаты показывают, что хэш-цепочечный реестр может обеспечивать защищённые от незаметного изменения записи аудита при небольших вычислительных затратах, тогда как его использование как сигнала консенсуса не повышает точность на LIAR.
Ключевые слова: оценка достоверности утверждений, дезинформация, блокчейн, классификация, дополненная поиском, DistilBERT, FAISS, набор данных LIAR, абляционный анализ, журнал аудита, модели-трансформеры.

1 Introduction

The rapid expansion of social media platforms, online news aggregators and political communication channels has intensified the challenge of distinguishing accurate from inaccurate or misleading content. Automated veracity-assessment systems are increasingly seen as a necessary component of the information ecosystem, as manual fact-checking cannot scale to the volume and velocity of digital content [1]. Related work on the automatic analysis

of political discourse and public speeches, including sentiment processing for low-resource languages such as Kazakh, shows that such content can be processed with domain ontologies and machine learning [2]. While transformer-based classifiers have advanced the state of the art on benchmark datasets, their deployment in high-stakes settings is limited by three shortcomings that motivate the present research.

First, predictions of neural classifiers are opaque and lack any reference to supporting evidence, making it difficult for human reviewers to validate or contest automated decisions [3]. Second, the probability that a classifier assigns to its prediction need not match its empirical accuracy [4], and there is no mechanism to take into account the accumulated history of similar, previously assessed claims. Third, centralized automated fact-checking systems usually keep no auditable trail of decisions, so that later modification of a recorded verdict cannot be detected.

Retrieval-Augmented Generation (RAG), introduced by Lewis et al. [5], partially addresses the first limitation by grounding predictions in evidence retrieved from a curated knowledge base. If the knowledge base contains only verified true statements, however, every retrieved neighbour carries the REAL label, and the retrieval score cannot point towards FAKE. A hash-chained ledger addresses the third limitation by recording decisions in a tamper-evident form [6]. We also ask whether the verdict history accumulated in such a ledger can serve as an empirical prior for borderline predictions.

This paper presents VeraChain, a framework that combines a fine-tuned DistilBERT classifier, a dual-index retrieval stage and a hash-chained audit ledger with a ledger-consensus step. The contributions are as follows. First, we describe a dual-index retrieval architecture with separate FAISS indices for REAL and FAKE reference statements and a probability-ratio score. Second, we describe a ledger-consensus step that, for borderline predictions, averages the stored probabilities of semantically similar ledger entries. Third, we evaluate the system on LIAR with a stepwise ablation (one index $\rightarrow$ two indices $\rightarrow$ ratio $\rightarrow$ ledger consensus) over five random seeds, with indices and ledger built on the training split and frozen for testing, exact paired tests and bootstrap intervals for accuracy and weighted F1, calibration metrics and latency percentiles. Fourth, we release the implementation, the per-statement predictions and the analysis scripts.

We use the term reliability in a specific and measurable sense, and assess four dimensions separately: (i) classification performance, measured by accuracy and weighted F1 with paired significance tests over several seeds; (ii) probability calibration, measured by the Brier score and the expected calibration error (ECE); (iii) evidence quality, i.e. whether retrieved statements support or refute the input claim rather than being merely topically related; and (iv) verdict reconstructability, i.e. whether an independent reviewer can recover the original verdict and its evidence from the ledger and detect any later alteration. Criteria (i), (ii) and (iv) are evaluated directly; for (iii) we report an automatic label-agreement proxy and real examples, since a blind human annotation was not carried out (Section 6).

Terminology. The LIAR labels are expert ratings of factual accuracy assigned by PolitiFact journalists; they do not establish whether a speaker intended to deceive. The task addressed in this paper is therefore the estimation of claim veracity, i.e. deciding whether a short political statement is predominantly accurate (REAL) or predominantly inaccurate (FAKE). The terms "disinformation detection" in the title and "RAG" are kept to situate the work in the corresponding research areas; we make no claim about the intent behind individual

statements, and the retrieval stage generates no text.

The remainder of the paper is organized as follows. Section 2 reviews related work. Section 3 presents the VeraChain architecture. Section 4 describes the data, baselines and evaluation protocol. Section 5 reports the ablation, the analysis of the consensus step, error rates, retrieval, calibration and latency. Section 6 discusses limitations, and Section 7 concludes.

2 Related work

Existing studies relevant to VeraChain span five areas: transformer-based disinformation detection, retrieval-augmented evidence grounding, blockchain applications for data integrity and fact verification, calibration of neural classifiers, and recent evidence-based and auditable fact-checking. Table 1 summarizes representative approaches.

Table 1: Comparison of VeraChain with existing approaches. Rows marked * were evaluated in this study on the same split (mean ± sd over five seeds, Section 5.1); the differences between them are not statistically significant

Study / method	Core technology	Dataset	Explainability	Audit trail	Accuracy (%)	Key limitation
CNN-based	CNN	LIAR	No	No	–	Weak contextual understanding
LSTM-based	LSTM	LIAR	No	No	–	Sequential dependency limits
BERT	Transformer	LIAR	Limited	No	–	No evidence retrieval
DistilBERT*	Lightweight transformer	LIAR	Limited	No	64.36 ± 1.25	No evidence retrieval
Single-index RAG*	DistilBERT + REAL-only index	LIAR	Yes	No	63.62 ± 0.88	Neighbours always REAL
Blockchain verification	Blockchain + rules	Various	Partial	Yes	–	Limited semantic modelling
VeraChain*	DistilBERT, dual index, ledger	LIAR	Yes	Yes	64.33 ± 1.36	No accuracy gain from consensus

CNN- and LSTM-based approaches provide baseline automatic classification capability but have limited contextual understanding. Transformer-based architectures such as BERT and DistilBERT substantially improve contextual modelling; however, they operate as black-box classifiers without explicit evidence retrieval or verifiable decision tracking. Standard RAG systems improve transparency by exposing the retrieved evidence, but a knowledge base containing only verified-real statements cannot provide a signal towards the FAKE class. Blockchain-based verification frameworks improve integrity guarantees and auditability, yet they generally do not use the recorded history at inference time. VeraChain combines these elements in one pipeline and evaluates each of them separately.

2.1 Transformer-based disinformation detection

The transformer architecture enables models to capture long-range dependencies and contextual representations [7]. BERT [7] has been fine-tuned for fake news classification on LIAR [8], FakeNewsNet [9] and FEVER [10]. DistilBERT [11], a distilled variant that retains 97 percent of BERT's performance with 40 percent fewer parameters, is suitable for resource-constrained deployment such as the RTX 3050 laptop GPU used in this study.

Accuracy on LIAR has remained low: text-only models evaluated on binary versions of LIAR typically reach 60 to 70 percent, for example from 64.4 percent for an SVM to 69.7 percent for CT-BERT in the evaluation of Iceland [12]. This ceiling reflects the ambiguity of political claims, where statements rated "mostly true", "half-true" or "barely true" share lexical and syntactic patterns. Wang [8] and, more recently, Alkomah and Sheldon [13] showed that adding speaker metadata or a speaker's fact-check history improves classification on LIAR, although this information is unavailable for novel claims.

Reported accuracies on binary LIAR are not directly comparable across studies, because studies differ in how the two intermediate labels are treated. Iceland [12] reports 69.7 percent for CT-BERT on a binary version of LIAR from which the "barely-true" and "half-true" statements were removed. In the present study all six labels are retained and the two intermediate labels are assigned to the binary classes (Table 2); these are the most ambiguous statements in the dataset, so the two figures should not be compared directly. Strong published models such as CT-BERT were not re-trained on our split (Section 6).

2.2 Retrieval-augmented evidence grounding

Retrieval-Augmented Generation was introduced by Lewis et al. [5] as a general framework for knowledge-intensive NLP tasks, combining dense passage retrieval with a sequence-to-sequence generator. In fact-checking, retrieval is used to surface supporting or refuting evidence before a verdict is produced [1]. Thorne et al. [10] showed on FEVER that evidence retrieval is a major bottleneck, as verdict accuracy drops substantially when gold evidence is replaced by retrieved evidence, and Guo et al. [1] emphasized that verdicts should be supported by the retrieved evidence. For a low-resource language, Rakhimova et al. [14] built question-answering datasets from official Kazakh legislation and evaluated fine-tuned large language models on them with automatic metrics and expert assessment. A design decision in retrieval-based classification is the composition of the index: a single index of verified-true statements returns only REAL neighbours. We therefore compare a REAL-only index, a mixed index with label voting and two class-specific indices (Section 5.1).

2.3 Blockchain for data integrity and fact verification

Blockchain technology, introduced by Nakamoto [6], offers an append-only ledger with cryptographic integrity guarantees. In information systems, blockchain-based provenance mechanisms have been proposed to record the origin and transformation history of digital content [15]. Devkota et al. [16] proposed VERA, a blockchain-backed framework that keeps auditable records of model runs and human decisions in news verification. Most existing work uses the ledger for provenance and tamper detection. In VeraChain, the ledger is additionally queried at inference time; whether this use improves accuracy is tested in Section 5.

2.4 Confidence calibration in neural classifiers

Guo et al. [4] showed that modern deep neural networks are often overconfident and proposed temperature scaling as a simple post-hoc calibration method. Subsequent work has explored label smoothing [17], Bayesian uncertainty quantification [18] and evidential deep learning [19]. The ledger-consensus step of VeraChain modifies individual borderline probabilities using similar past verdicts; we measure its effect on calibration and compare it with temperature scaling (Section 5.5).

2.5 Recent developments in evidence-based and auditable fact-checking

Since 2023, research on automated fact-checking has moved towards evaluating evidence, uncertainty and generalization in addition to verdict accuracy. Iceland [12] showed that transformer detectors, including CT-BERT, achieve high in-distribution accuracy on LIAR but degrade sharply on out-of-sample data. Pelrine et al. [20] studied GPT-4 for misinformation mitigation and argued that reliable systems must address generalization and uncertainty. Liu et al. [21] proposed TELLER, a framework targeting explainability, generalizability and controllability of fake news detection. The AVeriTeC shared task [22] scores the retrieved evidence together with the verdict, so that a correct label supported by inadequate evidence is not credited. Xu and Yan [23] showed that the evaluation of LLM-based fake news detectors can be inflated by data contamination and proposed temporally controlled evaluation. Alkomah and Sheldon [13] demonstrated that a speaker's prior fact-check history improves LIAR classification and transfers to LIAR2. VeraChain differs from VERA [16] in that its ledger is not only an audit record but is also queried at inference time as a source of historical verdicts for borderline predictions.

3 System architecture and methodology

3.1 System overview

VeraChain processes an input statement and produces a binary verdict (FAKE or REAL), a probability $f_{\text{final}}(x) = P(\text{REAL} \mid x)$ and a ledger record. Given an input statement x, the system first computes its SHA-256 hash and queries the hash index of the ledger. If an exact match is found, the stored verdict is returned. Otherwise the statement is processed by the DistilBERT classifier, the dual-index retrieval stage, the optional ledger-consensus step, and finally appended to the ledger. Figure 1 gives a schematic overview.

The ledger can be used in two modes. In the *frozen* mode, used for the main evaluation, the consensus search and the duplicate cache see only the entries that existed when the ledger was frozen (here: the training statements); new verdicts are still appended for audit but never influence later predictions. In the *online* mode, which corresponds to deployment and to the protocol of the earlier version of this paper, every new verdict becomes visible to later statements.

3.2 DistilBERT classifier

The base classifier is `distilbert-base-uncased` fine-tuned for binary sequence classification on the LIAR training split. Each statement is truncated to 128 tokens, and a linear head on

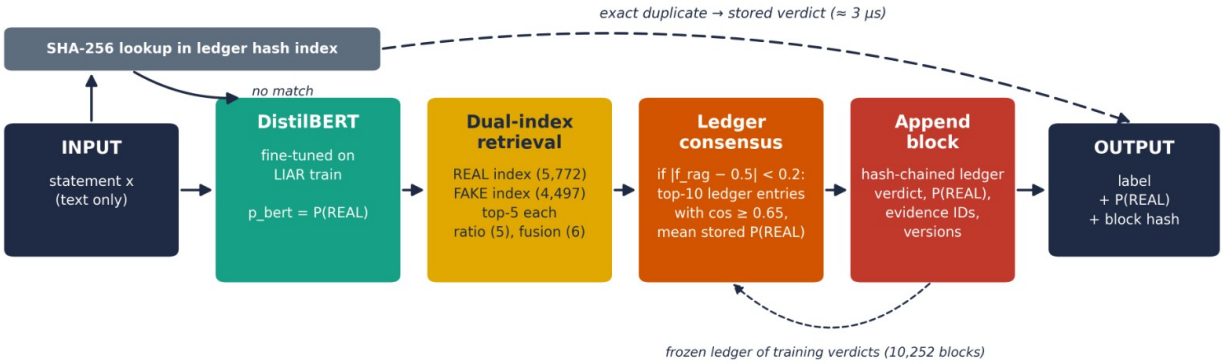


Figure 1: VeraChain architecture. A statement is first looked up by its SHA-256 hash; an exact duplicate returns the stored verdict (dashed line). Otherwise it passes DistilBERT, dual-index retrieval with fusion, the ledger-consensus step (borderline statements only, using the frozen ledger of training verdicts) and is appended to the hash-chained ledger

the [CLS] representation produces two logits:

$$[\text{logit}_{\text{FAKE}}, \text{logit}_{\text{REAL}}] = \text{DistilBERT}(x; \theta), \tag{1}$$

$$p_{\text{bert}}(x) = P_{\text{bert}}(\text{REAL} \mid x) = \text{softmax}([\text{logit}_{\text{FAKE}}, \text{logit}_{\text{REAL}}])_{\text{REAL}}. \tag{2}$$

Training uses AdamW [24] with weight decay $\lambda = 0.01$ on weight matrices and $\lambda = 0$ on biases and LayerNorm parameters, linear warm-up over 10 percent of the steps followed by linear decay, gradient clipping at norm 1.0, five epochs, batch size 32 and learning rate 2×10^{-5}; the checkpoint with the highest validation accuracy is kept. The original model (seed 42) was trained on an RTX 3050 laptop GPU with mixed precision; four additional models (seeds 1–4) were trained with the identical recipe on a CPU server (Section 4.3).

3.3 Dual-index retrieval

The knowledge base is built from the LIAR training split as two FAISS `IndexFlatIP` indices [25]: one with the sentence embeddings of the 5,772 REAL-labelled training statements and one with the 4,497 FAKE-labelled statements. Embeddings are produced by the `all-MiniLM-L6-v2` sentence transformer [26] and L2-normalized, so that the inner product equals the cosine similarity. For a query x, the top-k ($k = 5$) neighbours are retrieved from each index, and

$$s_r(x) = \frac{1}{k} \sum_{i=1}^{k} \cos\big(\text{emb}(x), \text{emb}(d_i)\big), \quad d_i \in \text{top-}k \text{ REAL}, \tag{3}$$

$$s_f(x) = \frac{1}{k} \sum_{j=1}^{k} \cos\big(\text{emb}(x), \text{emb}(d_j)\big), \quad d_j \in \text{top-}k \text{ FAKE}, \tag{4}$$

$$P_{\text{RAG}}(\text{REAL} \mid x) = \frac{s_r(x)}{s_r(x) + s_f(x)}. \tag{5}$$

$P_{\mathrm{RAG}}(\mathrm{REAL} \mid x)$ is below 0.5 whenever $s_f(x) > s_r(x)$ and approaches zero only when $s_r(x) \ll s_f(x)$. Because both mean similarities are of similar magnitude for MiniLM embeddings, the ratio stays close to 0.5: on the test set it lies between 0.386 and 0.593, with 90 percent of the values in $[0.467, 0.547]$ (Figure 7). Equation (5) therefore provides a directional but weak signal; its ranking quality is measured in Section 5.4.

The retrieval stage generates no text: its output is the scalar $P_{\mathrm{RAG}}(\mathrm{REAL} \mid x)$ together with the identifiers of the five best-matching statements from each index, which are stored in the ledger as the evidence for the verdict. We use the term RAG in the sense of retrieval-augmented classification. Retrieval measures semantic similarity, not evidential support: a retrieved statement contributes through its veracity label, and the system does not determine whether it supports, refutes or is merely topically related to the input claim. A real example of a misleading match is given in Section 5.4.

3.4 Confidence fusion

The DistilBERT probability and the retrieval score are fused linearly:

$$f_{\mathrm{rag}}(x) = \alpha\, p_{\mathrm{bert}}(x) + (1-\alpha)\, P_{\mathrm{RAG}}(\mathrm{REAL} \mid x), \qquad \alpha = 0.6,$$
$$\mathrm{label}_{\mathrm{rag}} = \begin{cases} \mathrm{REAL}, & f_{\mathrm{rag}}(x) \geq 0.5, \\ \mathrm{FAKE}, & \text{otherwise.} \end{cases} \tag{6}$$

The weight $\alpha = 0.6$ was set manually during development. A search over α on the validation split, reported in Section 5.6, selects $\alpha = 0.7$.

3.5 Blockchain audit layer

The ledger is an append-only chain of blocks implemented in pure Python with the `hashlib` standard library. Each block B_i stores: the index i; an ISO 8601 timestamp t_i; the SHA-256 hash of the statement text h_i; the verdict $v_i \in \{\mathrm{FAKE}, \mathrm{REAL}\}$; the value $c_i = f_{\mathrm{final}}(x_i) \in [0,1]$, i.e. the final probability of the REAL class written by the model ($v_i = \mathrm{REAL}$ if and only if $c_i \geq 0.5$); the identifiers of the ten retrieved statements (five from each index, prefixed R: and F:); the short hashes of the classifier checkpoint and of the knowledge-base indices; the index of the corrected block, if any; an optional text note; the hash of the preceding block H_{i-1}; and the block's own hash

$$H_i = \text{SHA-256}\big(i \,\|\, t_i \,\|\, h_i \,\|\, v_i \,\|\, c_i \,\|\, \mathrm{sources}_i \,\|\, m_i \,\|\, q_i \,\|\, r_i \,\|\, \mathrm{note}_i \,\|\, H_{i-1}\big), \tag{7}$$

where m_i and q_i are the model and knowledge-base version hashes, r_i is the index of the block corrected by B_i and note_i is an optional text note, e.g. the reason for a correction (both are empty for an ordinary verdict). The genesis block B_0 has `prev_hash` equal to 64 zero characters. The `verify_chain()` procedure recomputes every block hash and checks the links:

$$\texttt{verify_chain}() = \mathrm{True} \iff \forall\, i \geq 1 :\ H_i = \text{SHA-256}(B_i \backslash \{H_i\}) \wedge B_i.\texttt{prev_hash} = H_{i-1}. \tag{8}$$

A hash index maps each statement hash to its block, so that an exact duplicate is answered from the stored verdict without running the classifier. Figure 2 shows the block structure. The stored value c_i is the model's own probability, not the gold PolitiFact label.

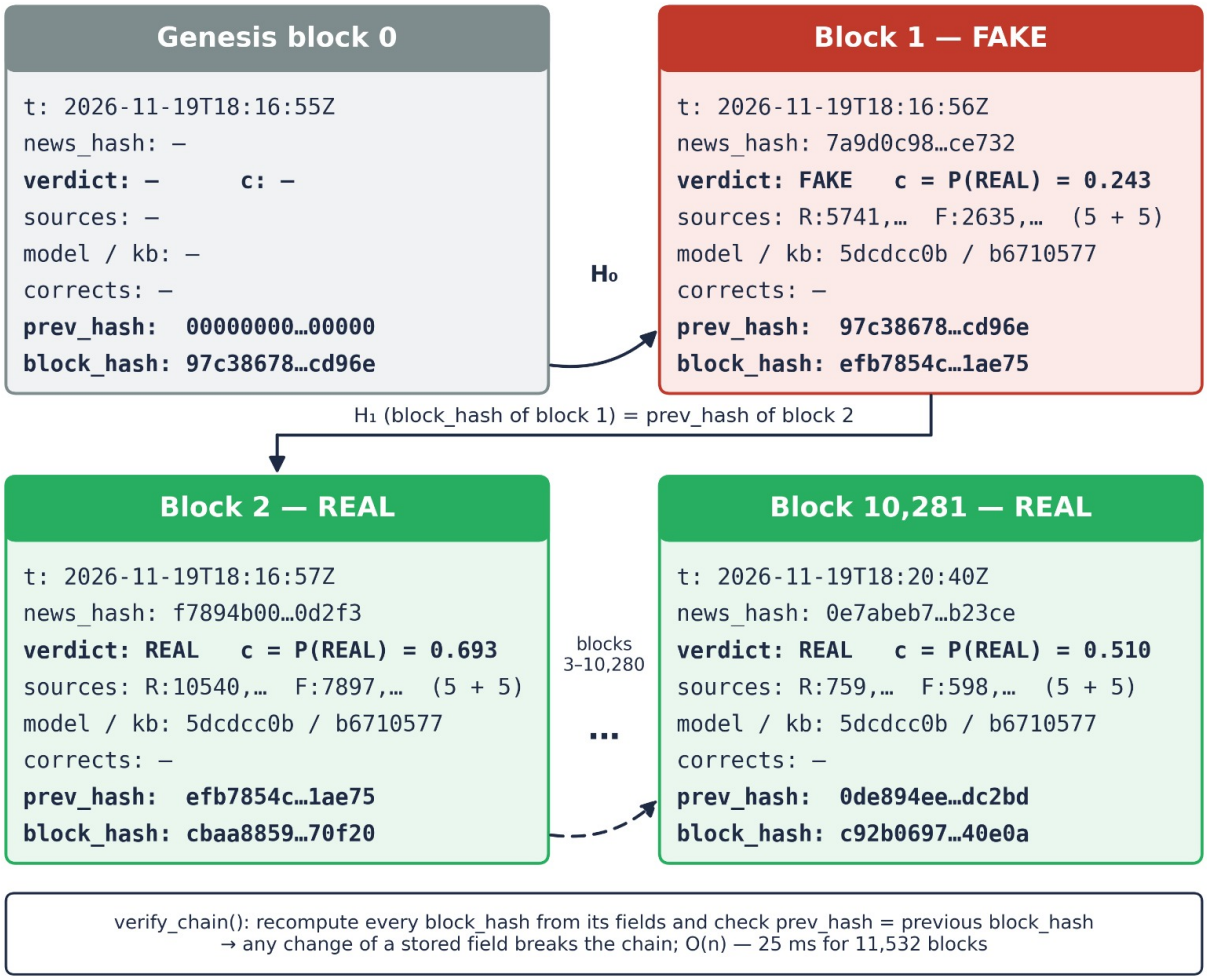


Figure 2: Ledger structure, drawn from real blocks of the seed-42 run: the genesis block, the first two training blocks and block 10,281, which records the corrected test case of Section 5.2. Each block stores the timestamp, the hash of the statement text, the verdict with $c_i = P(\text{REAL})$, the evidence identifiers (five from each index), the model and knowledge-base versions, the reference to a corrected block (if any), the hash of the preceding block and its own hash; `verify_chain()` detects alterations in $O(n)$

3.6 Ledger-consensus step

The consensus step uses a FAISS `IndexFlatIP` over the sentence embeddings of the ledger entries (excluding the genesis block). It is triggered for borderline predictions,

$$|f_{\text{rag}}(x) - 0.5| < \delta, \qquad \delta = 0.20, \tag{9}$$

i.e. for $f_{\text{rag}}(x) \in (0.30, 0.70)$, provided the ledger holds at least five entries. Among the $K = 10$ most similar entries, those with similarity of at least τ are kept:

$$N_{\text{sim}}(x) = \{B_j \in \text{top-}K : \ \cos(\text{emb}(x), \text{emb}(B_j.\text{text})) \geq \tau\}, \qquad \tau = 0.65. \tag{10}$$

If $N_{\text{sim}}(x)$ is non-empty, the final probability and label are

$$f_{\text{final}}(x) = \beta\, f_{\text{rag}}(x) + (1-\beta)\, C_{\text{chain}}(x), \ \ \beta = 0.6, \ \ \text{label}_{\text{final}} = \begin{cases} \text{REAL}, & f_{\text{final}}(x) \geq 0.5, \\ \text{FAKE}, & \text{otherwise}, \end{cases} \tag{11}$$

where

$$C_{\text{chain}}(x) = \frac{1}{|N_{\text{sim}}(x)|} \sum_{B_j \in N_{\text{sim}}(x)} c_j. \tag{12}$$

Otherwise f_{rag} and $\text{label}_{\text{rag}}$ are returned unchanged.

Because each c_j is a probability of the REAL class, Equation (12) takes the neighbours' verdicts into account: a neighbour classified as FAKE ($c_j < 0.5$) pulls the prediction towards FAKE and a neighbour classified as REAL pulls it towards REAL. (The earlier version of this paper stated that c_j is the probability of the predicted class and that the step can only change FAKE into REAL; this description did not match the implementation and is corrected here.) From Equation (11), the consensus step changes a label if and only if

$$|f_{\text{rag}}(x) - 0.5| < \frac{1-\beta}{\beta}\, |C_{\text{chain}}(x) - 0.5| \quad \text{and} \quad \text{sign}(C_{\text{chain}}(x) - 0.5) \neq \text{sign}(f_{\text{rag}}(x) - 0.5), \tag{13}$$

so it can move a prediction only towards the side of C_{chain} and, with $\beta = 0.6$, only when $|f_{\text{rag}} - 0.5| < \frac{2}{3}|C_{\text{chain}} - 0.5| \leq \frac{1}{3}$. This one-sided property is verified on all changed test predictions in Section 5.2.

The ledger is pre-populated by running the pipeline over the 10,269 training statements in file order; 17 of them repeat an earlier statement verbatim and are answered from the hash index, so the ledger holds 10,252 training blocks. Each block stores the model's own prediction for that statement, not the gold label; the model's verdict agrees with the gold binary label on 83.1 percent of these blocks. The parameters δ, τ, β and K were set manually during development; a validation search is reported in Section 5.6.

3.7 Audit procedure and verdict reconstruction

The ledger is written by the operator of the system: each block is appended by the classification service when a verdict is issued. It can be verified by anyone who holds a copy of the ledger, or only the hash of a block published earlier, by running `verify_chain()`.

Suppose that a verdict recorded in block B_i is later disputed. Given the statement text and a copy of the ledger, a reviewer (i) recomputes the hash of the text and locates B_i through the hash index; (ii) reads the verdict, the stored probability and the time of issue; (iii) retrieves the evidence statements through their identifiers; and (iv) runs `verify_chain()` to confirm that neither B_i nor any earlier block has been altered. If the stored verdict or an evidence identifier in B_i had been modified after recording, the recomputed hash of B_i would no longer match H_i and the link from B_{i+1} would break. A real case reconstructed in this way is shown in Section 5.2.

The revised implementation records in every block a short hash of the classifier checkpoint and of the knowledge-base indices, so that the reviewer can identify the exact model state that produced a verdict, and supports explicit correction blocks that reference the corrected block instead of modifying it. The ledger runs on a single node, and the operator who writes the blocks could also recompute the entire chain. Unless the hash of the latest block is periodically published to an external, independently controlled location, VeraChain offers the same guarantees as an append-only log with hash chaining: it makes alterations detectable by anyone holding an earlier copy of the chain or of its latest hash, but it does not prevent them. Moreover, the consensus search in Equation (10) operates on embeddings of the statement texts, whereas each block stores only the hash of its text; the texts and embeddings used by the consensus step are kept outside the chain and are not protected by it.

4 Experimental setup

4.1 Dataset

Experiments use the LIAR dataset [8]: 12,836 short political statements from PolitiFact.com, rated on a six-level scale (pants-fire, false, barely-true, half-true, mostly-true, true). We use the official files `train.tsv`, `valid.tsv` and `test.tsv` without modification; the test set is the official `test.tsv`. The labels are binarized as FAKE $\leftarrow$ {pants-fire, false, barely-true} and REAL $\leftarrow$ {half-true, mostly-true, true}. Table 2 gives the counts per label and split. The training file contains 10,269 statements, of which 10,252 are distinct texts: 17 statements repeat an earlier statement verbatim. All 10,269 statements are used to train the classifier and to build the two indices; the ledger holds 10,252 training blocks because repeated texts are answered from the hash index (Section 3.6). (The earlier version of this paper reported the number of distinct texts, 10,252, as the training-set size and the corresponding class counts as 5,370/4,882; the correct counts are given in Table 2.) Four test statements also occur verbatim in the training split. Only the statement text is used; speaker, subject and context fields are excluded.

4.2 Configurations and baselines

To separate the contributions of the components, all configurations share the same DistilBERT checkpoint, the same encoder (`all-MiniLM-L6-v2`), the same corpus (LIAR training split), the same number of retrieved items ($k = 5$ per index) and the same fusion weight $\alpha = 0.6$; none of them was tuned on the test set (equal, zero tuning budget). The configurations form a stepwise chain:

Table 2: Six LIAR labels, their binary class and the number of statements per split (official files)

Binary class	LIAR label	Train	Validation	Test
FAKE	pants-fire	842	116	92
	false	1,998	263	250
	barely-true	1,657	237	214
REAL	half-true	2,123	248	267
	mostly-true	1,966	251	249
	true	1,683	169	211
	FAKE / REAL	4,497 / 5,772	616 / 668	556 / 727
	Total (share REAL)	10,269 (56.2%)	1,284 (52.0%)	1,283 (56.7%)

C0 DistilBERT alone;

C1 + one index of REAL statements, retrieval score $s_r(x)$ ("single-index RAG");

C1v + one mixed index of all training statements, retrieval score = share of REAL labels among the top-5 (label vote);

C2 + two indices, score $0.5 + (s_r - s_f)/2$ (difference instead of ratio);

C3 + ratio, Equation (5) (dual-index RAG without the ledger);

C4 + ledger consensus, ledger with model verdicts built on the training split and frozen for testing (VeraChain, main protocol);

C5 as C4, but the ledger stores the gold labels of the training statements ($c_j \in \{0, 1\}$);

ON as C4, but the ledger is online: test verdicts are appended and visible to later test statements (protocol of the earlier version).

In addition, a TF-IDF + logistic regression baseline (50,000 uni- and bigram features, $C = 1.0$) is reported for reference. In the earlier version of this paper, the row labelled "RAG (single-idx)" corresponded to configuration C3, i.e. to the dual-index system without the consensus step; the single-index configuration C1 was not evaluated.

4.3 Evaluation protocol and implementation

Both FAISS indices and the ledger are built from the training split only; no validation or test label is used to construct any index or ledger entry. For the main evaluation (C4), the ledger is frozen after pre-population, and the 1,283 test statements are processed one at a time in the order of the test file. Statistics are computed from the per-statement predictions: exact McNemar tests on the discordant pairs, 95 percent paired bootstrap intervals (10,000 resamples) for the differences in accuracy and weighted F1, and mean ± standard deviation over five DistilBERT seeds (42, 1, 2, 3, 4). Seed 42 is the checkpoint of the earlier version, and its predictions reproduce the earlier results exactly (1,283 of 1,283 identical labels). Calibration is measured by the Brier score, the negative log-likelihood and the top-label ECE with 15 equal-width bins; temperature scaling is fitted on the validation split.

Inference and latency measurements were carried out on a laptop with an NVIDIA RTX 3050 Laptop GPU (4 GB VRAM) and 16 GB RAM, with PyTorch, HuggingFace Transformers [27], sentence-transformers [26] and `faiss-cpu`. The original pipeline for one seed (training, index construction, ledger pre-population, evaluation and figures) takes about 45 minutes on this laptop; the four additional DistilBERT models were trained in parallel on a 56-thread CPU server. All per-statement predictions and analysis scripts are released with the paper.

5 Experimental results and analysis

5.1 Stepwise ablation

Table 3 and Figure 3 report accuracy and weighted F1 for each configuration, for seed 42 and as mean ± sd over five seeds; Table 4 reports the change at each step.

Table 3: Stepwise ablation on the LIAR test set (n = 1,283). Seed 42 and mean ± sd over five seeds

Configuration	Acc. (s42)	wF1 (s42)	Acc. mean ± sd	wF1 mean ± sd
TF-IDF + LR (reference)	0.6235	0.6086	–	–
C0 DistilBERT	0.6485	0.6416	0.6436 ± 0.0125	0.6359 ± 0.0099
C1 + one index (REAL only)	0.6376	0.6283	0.6362 ± 0.0088	0.6258 ± 0.0054
C1v + one mixed index, label vote	0.6345	0.6227	0.6379 ± 0.0090	0.6268 ± 0.0075
C2 + two indices (difference)	0.6461	0.6388	0.6449 ± 0.0146	0.6367 ± 0.0116
C3 + ratio, Eq. (5)	0.6477	0.6402	0.6444 ± 0.0143	0.6362 ± 0.0114
C4 + ledger consensus, frozen (VeraChain)	0.6524	0.6434	0.6433 ± 0.0136	0.6335 ± 0.0110
C5 + ledger consensus, gold verdicts	0.6508	0.6445	0.6421 ± 0.0116	0.6348 ± 0.0108
ON + ledger consensus, online	0.6524	0.6434	0.6436 ± 0.0130	0.6338 ± 0.0103

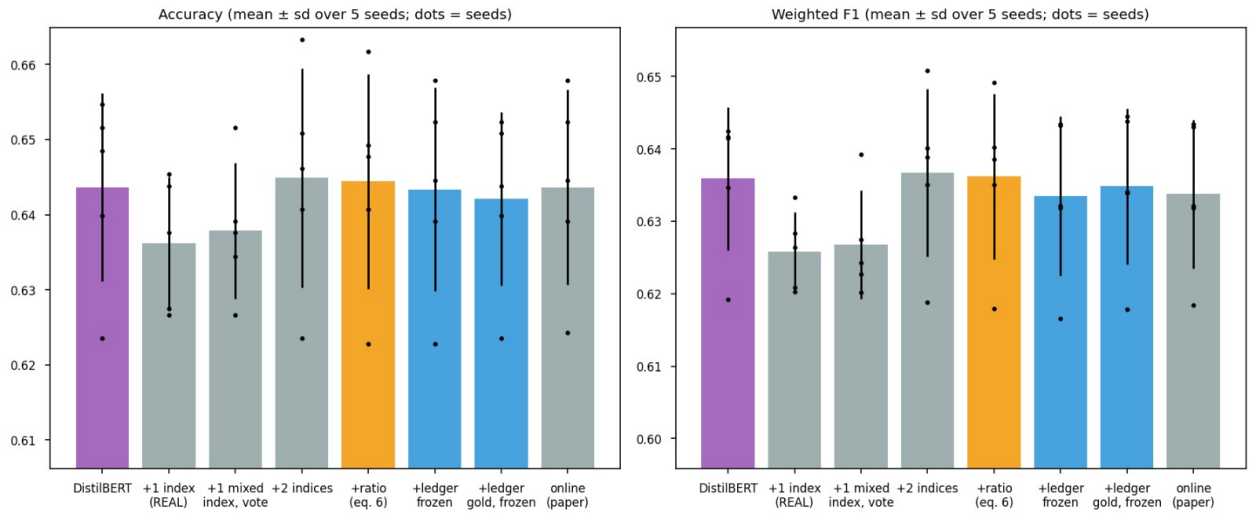


Figure 3: Accuracy and weighted F1 of the ablation configurations (bars: mean over five seeds, error bars: ±1 sd, dots: individual seeds)

Table 4: Change at each ablation step (percentage points). Seed 42: difference with paired 95% bootstrap interval and exact McNemar test (b/c = discordant pairs). Five seeds: mean ± sd of the difference and number of seeds with a positive change

Step	ΔAcc s42 [95% CI]	ΔwF1 s42 [95% CI]	b/c, p	ΔAcc, 5 seeds	> 0
C0→C1	−1.09 [−2.18, 0.00]	−1.33 [−2.47, −0.17]	33/19, 0.070	-0.75 ± 0.64	1/5
C1→C2	+0.86 [−0.16, 1.87]	+1.04 [0.01, 2.09]	16/27, 0.126	$+0.87 \pm 0.85$	4/5
C2→C3	+0.16 [0.00, 0.39]	+0.14 [0.00, 0.35]	0/2, 0.500	-0.05 ± 0.13	1/5
C3→C4	+0.47 [−0.39, 1.33]	+0.32 [−0.58, 1.18]	12/18, 0.362	-0.11 ± 0.37	1/5
C3→C5	+0.31 [−1.01, 1.64]	+0.43 [−0.92, 1.78]	36/40, 0.731	-0.23 ± 0.50	2/5
C4→ON	0.00 [−0.23, 0.23]	0.00 [−0.27, 0.27]	1/1, 1.000	$+0.03 \pm 0.07$	1/5
C0→C4	+0.39 [−0.70, 1.48]	+0.18 [−0.94, 1.27]	21/26, 0.560	-0.03 ± 0.43	2/5

Three observations follow. First, the seed-to-seed standard deviation of DistilBERT accuracy (1.25 points) is larger than every difference between configurations, and none of the paired tests for seed 42 is significant at the 5 percent level. Second, the only consistent effect is associated with the index design: a single REAL-only index lowers accuracy relative to DistilBERT alone (C0→C1, −0.75 points on average, lower in four of five seeds), and adding the FAKE index recovers this loss (C1→C2, +0.87 points, higher in four of five seeds). This is consistent with the real-label bias of a REAL-only index: on the test set, its retrieval score ranks REAL above FAKE statements with an AUC of only 0.578. The dual-index design does not, however, improve on DistilBERT alone (C0→C3, $+0.08 \pm 0.37$ points). A mixed single index with label voting (C1v) is not better than the REAL-only index. Third, the ratio of Equation (5) does not improve on the simple difference (C2→C3, −0.05 points), and the ledger-consensus step does not improve accuracy on average, either with model verdicts (C3→C4, −0.11 points) or with gold verdicts (C3→C5, −0.23 points). The improvement of 0.47 points obtained with seed 42 is not reproduced by the other seeds (changes of −0.47, −0.39, 0.00 and −0.16 points). Freezing the ledger changes only two test predictions relative to the online protocol for seed 42 (one in each direction, C4→ON), so the results of the earlier version were not driven by test verdicts leaking into the ledger.

5.2 What the consensus step does

For seed 42 and the frozen ledger, 865 of the 1,283 test statements (67.4 percent) fall in the borderline zone $f_{\text{rag}} \in (0.30, 0.70)$; four statements are exact duplicates of training statements and are answered from the hash index. For 309 borderline statements at least one ledger entry has similarity $\geq \tau = 0.65$ (on average 3.1 entries), and the consensus step is applied. It changes 30 labels: 18 incorrect predictions become correct and 12 correct predictions become incorrect (net +6); 23 changes are from FAKE to REAL and 7 from REAL to FAKE (Figure 4). Under the online protocol of the earlier version, the step is applied to 320 statements and changes 32 labels (19 corrected, 13 introduced; 24 FAKE→REAL and 8 REAL→FAKE). All changed predictions satisfy the one-sided property of Section 3.6. The preference for REAL reflects the content of the ledger: 63.2 percent of the training blocks carry a REAL verdict, while 56.2 percent of all 10,269 training statements are REAL. With gold labels in the ledger (C5), the step changes 76 labels (40 corrected, 36 introduced).

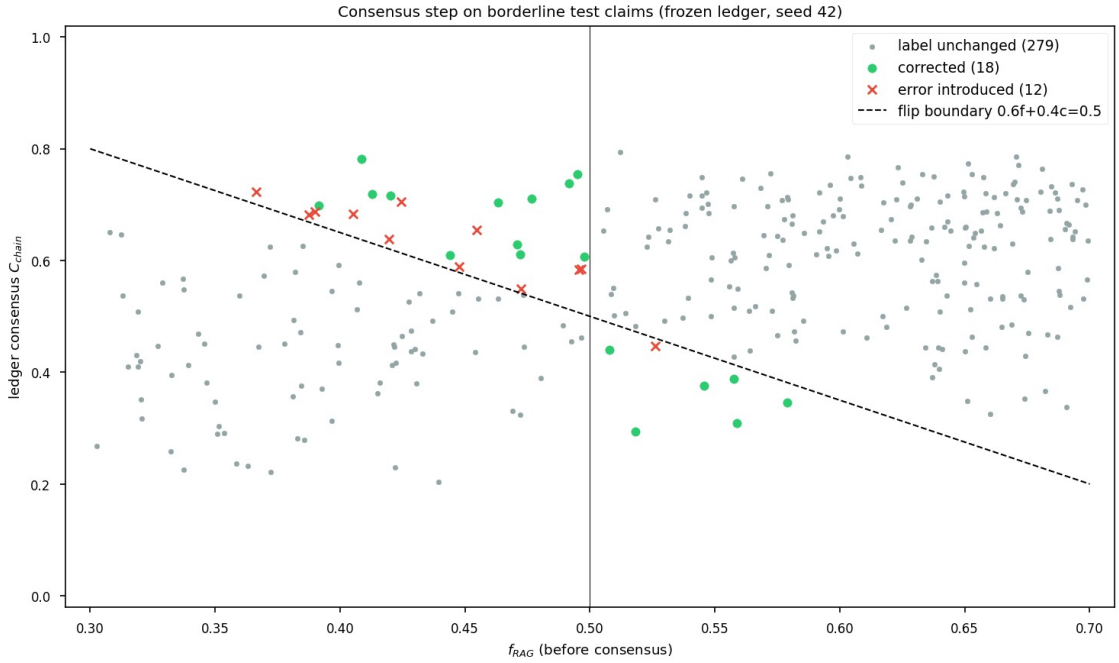


Figure 4: Consensus step on the 309 borderline test statements for which it is applied (seed 42, frozen ledger). Horizontal axis: f_{rag} before consensus; vertical axis: C_{chain}. The dashed line is the decision boundary $0.6\, f_{\text{rag}} + 0.4\, C_{\text{chain}} = 0.5$; labels change only for points between this line and $f_{\text{rag}} = 0.5$

A real corrected case. For the test statement LIAR ID 1051 (speaker mike-pence, gold label true): "Contends that President Obama literally said (if) his cap-and-trade proposals were to pass, that utility rates, his words now, would, necessarily skyrocket", DistilBERT (model version `5dcdcc0b9b8f`, knowledge base `b6710577c27a`) gives $p_{\text{bert}} = 0.401$ and the retrieval score is $P_{\text{RAG}}(\text{REAL} \mid x) = 0.508$, so $f_{\text{rag}} = 0.444$ (FAKE). The consensus search finds one ledger entry above the threshold: block 1920 (training statement ID 759, gold half-true, "Under Barack Obama's plan, the government would spend $1-trillion more"), similarity 0.655, stored $c_j = 0.609$ (REAL). Hence $f_{\text{final}} = 0.6 \cdot 0.444 + 0.4 \cdot 0.609 = 0.510$ (REAL), which is correct. The verdict is recorded in block 10281 (block hash beginning `c92b06975f7f6e89`), with the ten evidence identifiers and both version hashes, and a reviewer can reconstruct it as described in Section 3.7. The case also shows the limitation of the step: the neighbour that corrected the prediction is only topically related to the claim and passes the threshold by a narrow margin.

5.3 Error profile

Table 5 compares the error rates of the dual-index system without the ledger (C3) and VeraChain (C4) for seed 42, with REAL as the positive class; Table 6 gives the exact confusion matrices computed from the per-statement predictions.

VeraChain lowers the FNR by 1.51 points and raises the FPR by 0.90 points (Figure 5), because the consensus step changes more predictions towards REAL than towards FAKE.

Table 5: False positive rate (FAKE classified as REAL) and false negative rate (REAL classified as FAKE), seed 42. Changes in percentage points and, in parentheses, relative to C3

Configuration	FPR	FNR	ΔFPR	ΔFNR
C0 DistilBERT	0.504	0.235	–	–
C1 single-index RAG	0.536	0.230	–	–
C3 dual-index RAG	0.511	0.231	–	–
C4 VeraChain	0.520	0.216	+0.90 p.p. (+1.8%)	−1.51 p.p. (−6.5%)

Table 6: Confusion matrices on the test set, seed 42 (REAL = positive class)

Configuration	TP	FN	FP	TN
C0 DistilBERT	556	171	280	276
C1 single-index RAG	560	167	298	258
C3 dual-index RAG	559	168	284	272
C4 VeraChain	570	157	289	267

VeraChain correctly classifies 570 of the 727 REAL statements (recall 78.4 percent) and 267 of the 556 FAKE statements (48.0 percent). The low recall on FAKE is consistent with the difficulty of LIAR, where "barely-true" statements share extensive lexical overlap with "half-true" and "mostly-true" statements. Figure 6 shows the confusion matrix of VeraChain.

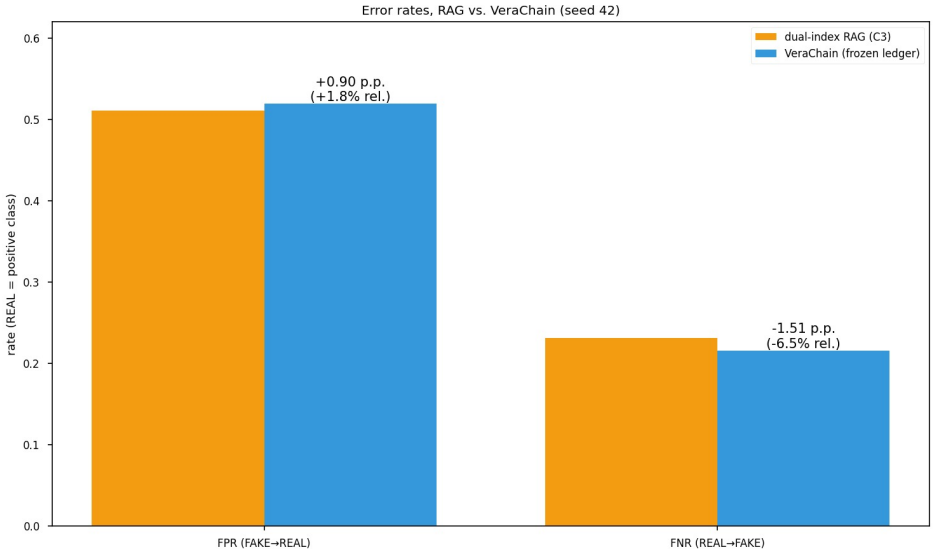


Figure 5: Error rates of the dual-index system without the ledger (C3) and VeraChain (C4), seed 42. The FPR increases by 0.90 points (+1.8% relative) and the FNR decreases by 1.51 points (−6.5% relative)

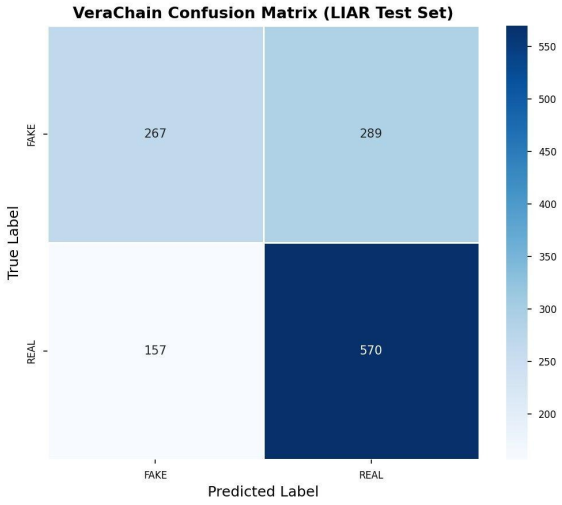


Figure 6: VeraChain confusion matrix on the test set (seed 42), REAL as the positive class: TP = 570, FN = 157, FP = 289, TN = 267

5.4 Retrieval quality

Figure 7 shows the distribution of $P_{\mathrm{RAG}}(\mathrm{REAL} \mid x)$ on the test set. Its mean is 0.513 for REAL and 0.502 for FAKE statements; only 1.4 percent of the values are below 0.45 and 4.0 percent above 0.55. As a ranking score for the REAL class, $P_{\mathrm{RAG}}(\mathrm{REAL} \mid x)$ reaches an AUC of 0.628, compared with 0.578 for the single-index score s_r, 0.609 for the mixed-index vote and 0.635 for the simple difference $s_r - s_f$.

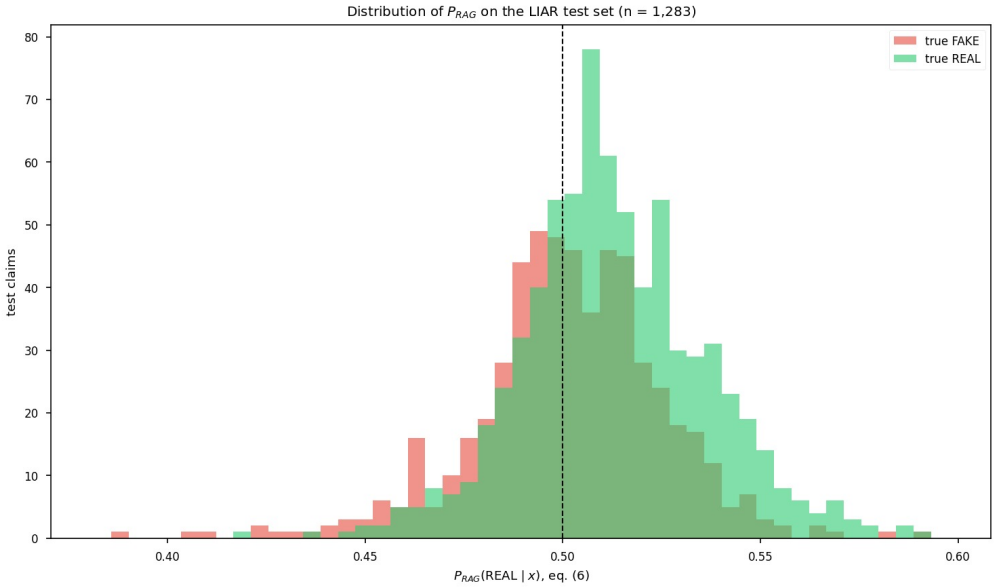


Figure 7: Distribution of $P_{\mathrm{RAG}}(\mathrm{REAL} \mid x)$ (Equation (5)) on the 1,283 test statements by true label

Whether retrieved statements are usable evidence was not annotated by human assessors.

As an automatic proxy we measure how often the gold label of a retrieved statement equals the gold label of the test claim. For the REAL-only index this share equals the proportion of REAL test claims (56.7 percent), since every neighbour is REAL. For the five best matches across both indices it is 56.3 percent, and for the single best match 58.3 percent. Only when the best match is very close does the label agree more often: 77.2 percent for the 79 claims with a neighbour of similarity ≥ 0.8 and 82.4 percent for the 17 claims with similarity ≥ 0.9. Six test claims have a nearest training neighbour with similarity ≥ 0.85 and the opposite label. For example, the test claim "Says Charlie Crist voted against raising the minimum wage" (LIAR ID 9612, half-true, REAL) retrieves the training claim "Says Charlie Crist voted against minimum wage when he was in the Senate" (ID 10088, false, FAKE) with cosine similarity 0.92; the two statements concern the same vote, but the retrieval score cannot tell whether the neighbour supports or contradicts the claim.

5.5 Calibration

Table 7 and Figure 8 report calibration on the test set for seed 42. Fusing the DistilBERT probability with the retrieval score lowers the Brier score from 0.238 to 0.224 and the ECE from 0.112 to 0.045, mainly because Equation (6) shrinks the overconfident DistilBERT probabilities towards 0.5. The consensus step does not improve calibration further (ECE 0.047). Temperature scaling of DistilBERT alone, fitted on the validation split ($T = 2.03$), gives an ECE of 0.031 at the same Brier score and leaves the predicted labels unchanged. The improved calibration of the fused systems is therefore a shrinkage effect that a single temperature parameter achieves at least as well.

Table 7: Calibration on the test set (seed 42). TS = temperature scaling fitted on the validation split

Model	T	Brier	ECE (15 bins)	NLL
C0 DistilBERT	–	0.2377	0.1116	0.6821
C3 dual-index RAG	–	0.2238	0.0453	0.6387
C4 VeraChain (frozen)	–	0.2242	0.0472	0.6395
ON VeraChain (online)	–	0.2241	0.0449	0.6392
C0 + TS	2.031	0.2243	0.0309	0.6396
C3 + TS	1.050	0.2238	0.0325	0.6386
C4 + TS	1.009	0.2242	0.0445	0.6394

Figure 9 shows the distribution of the final probability by true label. The mean f_{final} is 0.52 for FAKE and 0.61 for REAL statements, and the two distributions overlap mainly between 0.40 and 0.70, i.e. largely inside the borderline zone $(0.30, 0.70)$ of Equation (9).

5.6 Validation tuning

The hyper-parameters of the earlier version were set manually. Selecting α on the validation split for C3 gives $\alpha = 0.7$ and a test accuracy of 64.61 percent (64.77 percent with $\alpha = 0.6$). A grid search over α, δ, τ, β and K for C4 (2,700 combinations, frozen ledger) selects $\alpha = 0.6$, $\delta = 0.20$, $\tau = 0.50$, $\beta = 0.6$, $K = 10$ with a validation accuracy of 65.26 percent; on the test

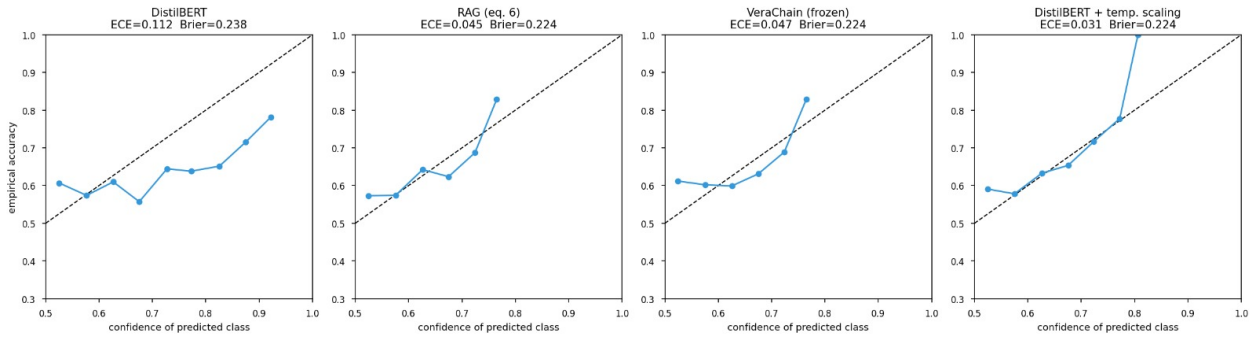


Figure 8: Reliability diagrams on the test set (seed 42): DistilBERT, dual-index RAG, VeraChain and DistilBERT with temperature scaling

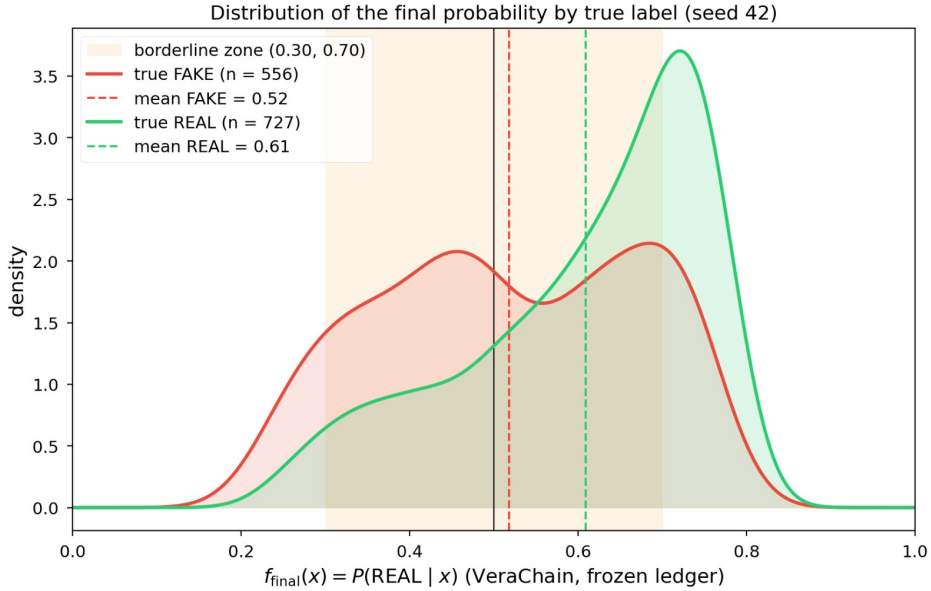


Figure 9: Distribution of the VeraChain probability $f_{\text{final}} = P(\text{REAL})$ by true label (seed 42, frozen ledger), with the borderline zone $(0.30, 0.70)$ of Equation (9) shaded. The overlap between 0.40 and 0.70 lies within this zone

set this gives 65.32 percent accuracy but a lower weighted F1 (0.6384 versus 0.6434), because the lower similarity threshold moves more predictions to REAL (FPR 0.565, FNR 0.180). With the default parameters, validation accuracy is 63.40 percent for C4, 63.24 percent for C3 and 63.24 percent for DistilBERT alone. Tuning on the validation split therefore does not change the conclusions of Section 5.1.

5.7 Duplicates, near-duplicates and novel claims

Four test statements are exact duplicates of training statements and are answered from the hash index; all four are classified correctly by every configuration. Thirteen further test statements have a training neighbour with cosine similarity ≥ 0.90 (near-duplicates); accuracy on them is 53.8 percent for DistilBERT, C3 and C4, and 61.5 percent for the single-index configuration. On the remaining 1,266 novel claims, accuracy is 64.85 percent for DistilBERT, 64.77 percent for C3 and 65.24 percent for C4. A date-ordered replay was not possible because the LIAR files contain no claim dates.

5.8 Latency

Table 8 reports per-statement processing times of the revised implementation, measured on the laptop described in Section 4.3 (RTX 3050 Laptop GPU, 4 CPU threads, no concurrent load, 20 warm-up statements). The ledger was pre-populated with the training statements (10,252 blocks, 223 s), frozen, and the 1,283 test statements were processed one at a time; every new block was written to a JSON-lines file with `fsync`. The predictions of this end-to-end run agree with the analysis of Section 5.1 on all 1,283 statements.

Table 8: Processing time per statement (milliseconds), seed 42, frozen ledger with 10,252 training blocks

Stage / system	n	Mean	Median	95th pct.
DistilBERT only	500	5.68	5.63	6.14
Dual-index RAG (C3, no ledger)	500	15.56	15.30	18.09
VeraChain, all test statements	1,283	17.72	17.50	20.85
DistilBERT + retrieval	1,279	16.68	16.41	19.55
consensus search (borderline only)	865	0.94	0.89	1.13
block append incl. disk write (`fsync`)	1,279	0.46	0.43	0.56
exact-duplicate lookup (hash index)	500	0.003	0.003	0.004

The ledger adds about 2 ms per statement to the dual-index RAG system (median 17.50 versus 15.30 ms): about 0.9 ms for the consensus search on borderline statements and 0.4 ms for appending the block and writing it to disk. Most of the time is spent on sentence embedding and retrieval, not on DistilBERT. An exact duplicate is answered from the hash index in about 3 μs. Verifying the whole chain of 11,532 blocks with `verify_chain()` takes 25 ms, reloading it from disk 0.1 s, and the ledger file occupies 7.2 MB; a modification of a stored verdict was detected by `verify_chain()` in a tampering test. The earlier version of this paper reported mean times of 14–15 ms for RAG and 35–40 ms for VeraChain from a less controlled measurement with the online ledger and without the disk write; the values in

Table 8 supersede them. These figures indicate the order of magnitude of the overhead on a single laptop GPU rather than a production benchmark.

5.9 Embedding space analysis

Figure 10 shows two-dimensional UMAP [28] projections of the sentence embeddings of 500 test statements. Real and fake statements do not form separate clusters, which is consistent with the linguistic similarity of political claims across truthfulness classes on LIAR [8]. Misclassified statements are spread over the whole embedding space rather than concentrated at cluster boundaries. Together with the low AUC of the retrieval scores (Section 5.4), this explains why similarity-based retrieval adds little to the classifier on this dataset.

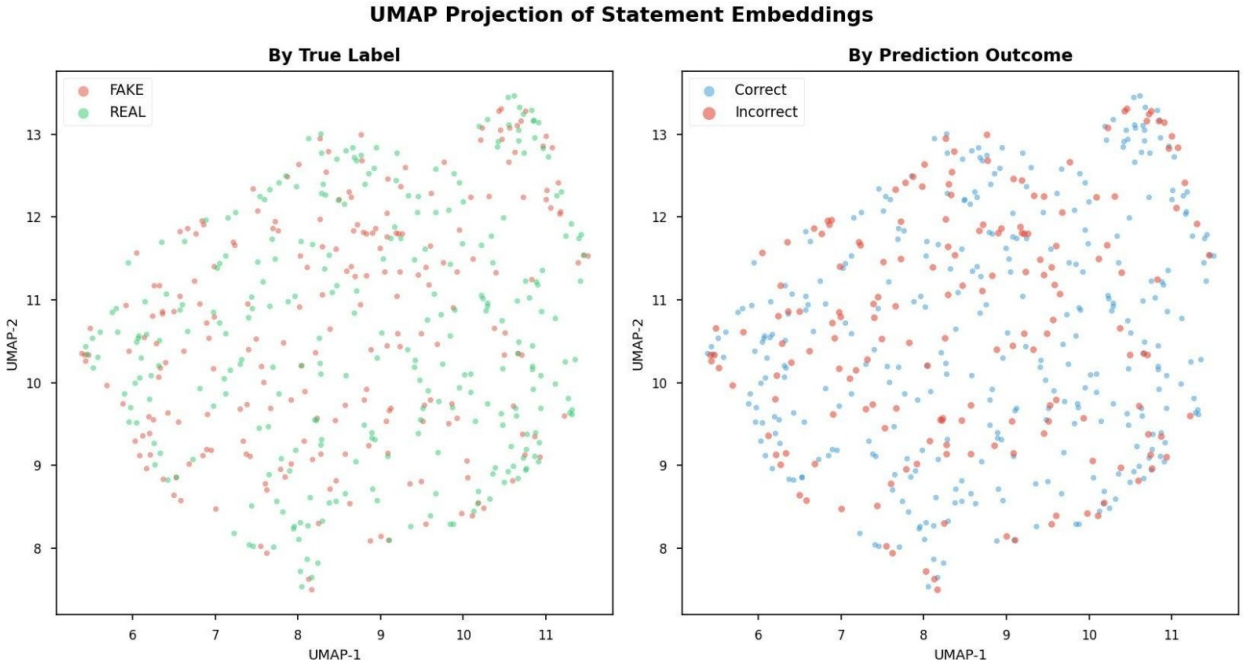


Figure 10: UMAP projection of 500 test statement embeddings. Left: coloured by true label (FAKE = red, REAL = green). Right: coloured by VeraChain outcome (correct = blue, incorrect = red)

6 Discussion

The ablation answers the question of which component contributes what. On LIAR, the fine-tuned classifier carries nearly all of the predictive signal. A REAL-only index harms accuracy, a second index of FAKE statements removes this harm, and neither the probability ratio nor the ledger consensus adds a consistent gain over the dual-index system. The gain reported in the earlier version of this paper for seed 42 is within the seed-to-seed variation. The earlier statement that the dual-index design "eliminates" the real-label bias is therefore replaced by the measured result: the second index removes the accuracy loss caused by a REAL-only

index but does not improve on the classifier alone. Similarly, the earlier Figure 9 ("component contribution") was replaced by the ablation in Figure 3.

The consensus step is applied to about a quarter of the test statements and changes about 2.5 percent of the labels, with almost as many errors introduced as corrected. Even gold labels in the ledger (C5) do not help, which indicates that the limitation lies in the retrieval of similar statements rather than in the noise of model verdicts: similar political statements frequently carry opposite ratings (Section 5.4). The step shifts the error profile towards REAL (lower FNR, higher FPR) because the ledger contains more REAL verdicts than the data. Retrieval fusion improves calibration, but temperature scaling of the classifier alone does so at least as well.

The value of the ledger therefore lies in auditability rather than accuracy: each verdict is stored with its evidence identifiers, model and knowledge-base versions and timestamp, any later alteration is detectable by holders of an earlier copy, and the median overhead per statement is small (Section 5.8).

Limitations. First, all results are on a single dataset of short political statements; strong published models such as CT-BERT or ConvBERT were not re-trained on our split, so the comparison with the literature in Section 2 is indicative only. Second, the evidence quality of retrieved statements was assessed by an automatic label-agreement proxy rather than by blind human annotation of supporting, refuting and merely topical evidence. Third, LIAR contains no claim dates, so a date-ordered replay of the ledger could not be performed. Fourth, five seeds vary only the classifier; the indices, encoder and fusion weights are fixed. Fifth, the ledger runs on a single node, provides tamper evidence only to parties holding an earlier copy of the chain or of its latest hash, and does not protect the texts and embeddings used by the consensus search; the consensus step also provides no benefit during cold start, when fewer than five entries are available. Future work will address external anchoring of the chain, human annotation of evidence, datasets with dated claims, and retrieval methods that model support and refutation rather than topical similarity. The importance of temporal and contextual consistency for early-stage fake news detection has also been emphasized in theory-driven research [29].

7 Conclusions

This paper presented VeraChain, a veracity-assessment framework that combines a fine-tuned DistilBERT classifier, a dual-index retrieval stage and a hash-chained audit ledger with a ledger-consensus step. A stepwise ablation over five seeds on the LIAR benchmark, with indices and ledger built on the training data and frozen for testing, shows that DistilBERT alone reaches 64.36 ± 1.25 percent accuracy; a REAL-only retrieval index lowers accuracy by 0.75 points, a second FAKE index recovers this loss, and the probability-ratio score and the ledger consensus add no consistent gain. With the original seed, VeraChain reaches 65.24 percent accuracy and a weighted F1 of 0.6434; the difference from the same system without the consensus step (0.47 points) is not statistically significant and is not reproduced by other seeds. Retrieval fusion improves calibration (ECE 0.045 versus 0.112), as does temperature scaling of the classifier alone (0.031). The median processing time of 17.5 milliseconds per statement on a single laptop GPU, including the write of the ledger entry to disk, indicates that a tamper-evident record of every verdict can be kept at a small cost.

The contributions of this work are therefore an auditable veracity-assessment pipeline in which every verdict is recorded with its evidence and model version, and a controlled evaluation showing which of its components affect accuracy on LIAR. Future directions include stronger backbones such as RoBERTa-large and DeBERTa-v3, retrieval that models support and refutation, datasets with dated claims for temporally ordered evaluation of the ledger, and external anchoring of the chain.

Authors contribution

A.T.: conceptualization, methodology, writing – original draft preparation. **S.N.**: formal analysis, supervision, writing – original draft preparation. **A.D.**: validation, writing – review and editing. **M.K.**: methodology, writing – review and editing. **L.K.**: investigation, data curation. All authors have read and agreed to the published version of the manuscript.

Acknowledgments

This research has been funded by the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan (Grant No. AP26197729).

References

[1] Guo, Z., M. Schlichtkrull, and A. Vlachos. 2022. "A Survey on Automated Fact-Checking." *Transactions of the Association for Computational Linguistics* 10: 178–206.

[2] Bekmanova, G., B. Yergesh, A. Ukenova, A. Omarbekova, A. Mukanova, and Y. Ongarbayev. 2023. "Sentiment Processing of Socio-political Discourse and Public Speeches." In *Computational Science and Its Applications – ICCSA 2023 Workshops*, Lecture Notes in Computer Science 14108, 191–205. Cham: Springer. https://doi.org/10.1007/978-3-031-37117-2_15.

[3] Popat, K., S. Mukherjee, A. Yates, and G. Weikum. 2018. "DeClarE: Debunking Fake News and False Claims Using Evidence-Aware Deep Learning." In *Proceedings of EMNLP*, 22–32.

[4] Guo, C., G. Pleiss, Y. Sun, and K. Q. Weinberger. 2017. "On Calibration of Modern Neural Networks." In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 1321–30.

[5] Lewis, P., E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela. 2020. "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks." *Advances in Neural Information Processing Systems* 33: 9459–74.

[6] Nakamoto, S. 2008. "Bitcoin: A Peer-to-Peer Electronic Cash System." https://bitcoin.org/bitcoin.pdf.

[7] Devlin, J., M.-W. Chang, K. Lee, and K. Toutanova. 2019. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." In *Proceedings of NAACL-HLT*, 4171–86.

[8] Wang, W. Y. 2017. "'Liar, Liar Pants on Fire': A New Benchmark Dataset for Fake News Detection." In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 422–26.

[9] Shu, K., D. Mahudeswaran, S. Wang, D. Lee, and H. Liu. 2020. "FakeNewsNet: A Data Repository with News Content, Social Context, and Spatiotemporal Information for Studying Fake News on Social Media." *Big Data* 8 (3): 171–88.

[10] Thorne, J., A. Vlachos, C. Christodoulopoulos, and A. Mittal. 2018. "FEVER: A Large-Scale Dataset for Fact Extraction and VERification." In *Proceedings of NAACL-HLT*, 809–19.

[11] Sanh, V., L. Debut, J. Chaumond, and T. Wolf. 2019. "DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter." In *5th Workshop on Energy Efficient Machine Learning and Cognitive Computing (NeurIPS Workshop)*.

[12] Iceland, M. 2023. "How Good Are SOTA Fake News Detectors." arXiv preprint arXiv:2308.02727.

[13] Alkomah, B., and F. Sheldon. 2026. "Credibility Context Improves Political Claim Classification on LIAR and Transfers to LIAR2." *Future Internet* 18 (7): 346. https://doi.org/10.3390/fi18070346.

[14] Rakhimova, D., A. Turarbek, V. Karyukin, A. Sarsenbayeva, and R. Alieyev. 2025. "Legal AI in Low-Resource Languages: Building and Evaluating QA Systems for the Kazakh Legislation." *Computers* 14 (9): 354. https://doi.org/10.3390/computers14090354.

[15] Huckle, S., and M. White. 2017. "Fake News: A Technological Approach to Proving the Origins of Content, Using Blockchains." *Big Data* 5 (4): 356–71. https://doi.org/10.1089/big.2017.0071.

[16] Devkota, P., et al. 2026. "VERA: A Blockchain-Backed, Auditable ML-Human Framework for News Verification and Misinformation Mitigation." In *Proceedings of IEEE ICSFT*. https://doi.org/10.1109/ICSFT66733.2026.11507787.

[17] Müller, R., S. Kornblith, and G. E. Hinton. 2019. "When Does Label Smoothing Help?" *Advances in Neural Information Processing Systems* 32.

[18] Wilson, A. G., and P. Izmailov. 2020. "Bayesian Deep Learning and a Probabilistic Perspective of Generalization." *Advances in Neural Information Processing Systems* 33: 4697–4708.

[19] Sensoy, M., L. Kaplan, and M. Kandemir. 2018. "Evidential Deep Learning to Quantify Classification Uncertainty." *Advances in Neural Information Processing Systems* 31.

[20] Pelrine, K., A. Imouza, C. Thibault, M. Reksoprodjo, C. Gupta, J. Christoph, J.-F. Godbout, and R. Rabbany. 2023. "Towards Reliable Misinformation Mitigation: Generalization, Uncertainty, and GPT-4." In *Proceedings of EMNLP*, 6399–6429. https://doi.org/10.18653/v1/2023.emnlp-main.395.

[21] Liu, H., W. Wang, H. Li, and H. Li. 2024. "TELLER: A Trustworthy Framework for Explainable, Generalizable and Controllable Fake News Detection." In *Findings of ACL 2024*, 15556–83. https://doi.org/10.18653/v1/2024.findings-acl.919.

[22] Schlichtkrull, M., Y. Chen, C. Whitehouse, Z. Deng, M. Akhtar, R. Aly, Z. Guo, C. Christodoulopoulos, O. Cocarascu, A. Mittal, J. Thorne, and A. Vlachos. 2024. "The Automated Verification of Textual Claims (AVeriTeC) Shared Task." In *Proceedings of the Seventh FEVER Workshop*, 1–26. https://doi.org/10.18653/v1/2024.fever-1.1.

[23] Xu, C., and N. Yan. 2025. "TripleFact: Defending Data Contamination in the Evaluation of LLM-driven Fake News Detection." In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 8808–23. https://doi.org/10.18653/v1/2025.acl-long.431.

[24] Loshchilov, I., and F. Hutter. 2019. "Decoupled Weight Decay Regularization." In *Proceedings of the 7th International Conference on Learning Representations (ICLR).*

[25] Johnson, J., M. Douze, and H. Jégou. 2021. "Billion-Scale Similarity Search with GPUs." *IEEE Transactions on Big Data* 7 (3): 535–47.

[26] Reimers, N., and I. Gurevych. 2019. "Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks." In *Proceedings of EMNLP-IJCNLP*, 3982–92.

[27] Wolf, T., L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, and J. Brew. 2020. "Transformers: State-of-the-Art Natural Language Processing." In *Proceedings of EMNLP: System Demonstrations*, 38–45.

[28] McInnes, L., J. Healy, and J. Melville. 2018. "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction." arXiv preprint arXiv:1802.03426.

[29] Zhou, X., A. Jain, V. V. Phoha, and R. Zafarani. 2020. "Fake News Early Detection: A Theory-Driven Model." *Digital Threats: Research and Practice* 1 (2): 1–25.

Information about authors:

Turarbek Assem – PhD, Associate Professor of the Faculty of Information Technology and Artificial Intelligence of the Farabi University (Almaty, Kazakhstan, email: asem.turarbek@kaznu.edu.kz).

Narbayeva Saltanat – Senior Lecturer of the Faculty of Information Technology and Artificial Intelligence of the Farabi University (Almaty, Kazakhstan, email: Narbaeva.salta@kaznu.kz).

Dokuz Ahmet Sakir – PhD, Associate Professor of the Nigde Omer Halisdemir University (Nigde, Turkey, email: adokuz@ohu.edu.tr).

Karimov Mukhit – Master student of the Faculty of Information Technology and Artificial Intelligence of the Farabi University (Almaty, Kazakhstan, email: karimov_ mukhit@live.kaznu.kz).

Kopbossyn Leila – Candidate of Technical Sciences, Assistant Professor of the Faculty of Information Technology and Artificial Intelligence of the Farabi University (Almaty, Kazakhstan, email: kopbosyn.leyliya@kaznu.kz).

Авторлар туралы мәлімет:

Турарбек Әсем Турарбекқызы – PhD, әл-Фараби атындағы Қазақ ұлттық университеті ақпараттық технологиялар және жасанды интеллект факультетінің доцент міндетін атқарушы (Алматы, Қазақстан, электрондық пошта: asem.turarbek@kaznu.edu.kz).

Нарбаева Салтанат Муратбековна – әл-Фараби атындағы Қазақ ұлттық университеті ақпараттық технологиялар және жасанды интеллект факультетінің аға оқытушысы (Алматы, Қазақстан, электрондық пошта: Narbaeva.salta@kaznu.kz).

Докуз Ахмет Шакир – PhD, Өмер Халисдемир атындағы Нигде университетінің доценті (Нигде, Түркия, электрондық пошта: adokuz@ohu.edu.tr).

Каримов Мухит Серикович – әл-Фараби атындағы Қазақ ұлттық университеті ақпараттық технологиялар және жасанды интеллект факультетінің магистранты (Алматы, Қазақстан, электрондық пошта: karimov_mukhit@live.kaznu.kz).

Көпбосын Лейля Серикбековна – техника ғылымдарының кандидаты, әл-Фараби атындағы Қазақ ұлттық университеті ақпараттық технологиялар және жасанды интеллект факультетінің ассистент профессоры (Алматы, Қазақстан, электрондық пошта: kopbosyn.leyliya@kaznu.kz).

Информация об авторах:

Турарбек Асем Турарбеккызы – PhD, исполняющий обязанности доцента факультета информационных технологий и искусственного интеллекта Казахского национального университета имени аль-Фараби (Алматы, Казахстан, электронная почта: asem.turarbek@kaznu.edu.kz).

Нарбаева Салтанат Муратбековна – старший преподаватель факультета информационных технологий и искусственного интеллекта Казахского национального университета имени аль-Фараби (Алматы, Казахстан, электронная почта: Narbaeva.salta@kaznu.kz).

Докуз Ахмет Шакир – PhD, Доцент Университета Нигде имени Омера Халисдемира (Нигде, Турция, электронная почта: adokuz@ohu.edu.tr).

Каримов Мухит Серикович – магистрант факультета информационных технологий и искусственного интеллекта Казахского национального университета имени аль-Фараби (Алматы, Казахстан, электронная почта: karimov_mukhit@live.kaznu.kz).

Копбосын Лейля Серикбековна – кандидат технических наук, ассистент профессора факультета информационных технологий и искусственного интеллекта Казахского национального университета имени аль-Фараби (Алматы, Казахстан, электронная почта: kopbosyn.leyliya@kaznu.kz).

Revised: October 19, 2026
Accepted: November 21, 2026