Sulla sinergia tra apprendimento automatico e filtri di Bloom

Giacomo Fumagalli

Relatore: Prof. Dario Malchiodi
Correlatore: Prof. Marco Frasca

Filtro di Bloom - Introduzione

- Un filtro di Bloom [1] è una struttura dati probabilistica, basata su funzioni di hash, spesso utilizzata per verificare l'appartenenza di un elemento a un dato insieme.
- Per definizione, un filtro di Bloom può generare solamente falsi positivi.
- Le operazioni che caratterizzano questa struttura dati sono inserimento e verifica dell'appartenenza di un elemento.

Filtro di Bloom - Operazioni

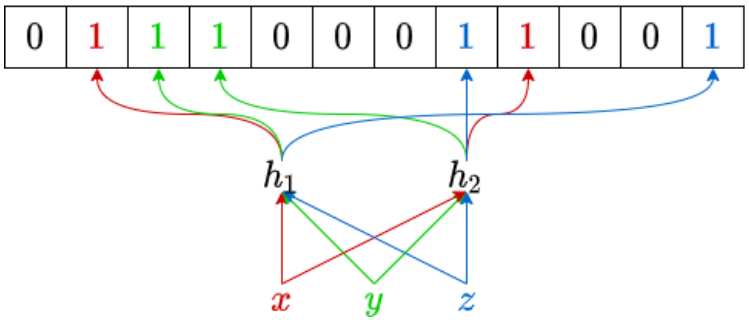


Figura: Esempio dell'operazione di inserimento.

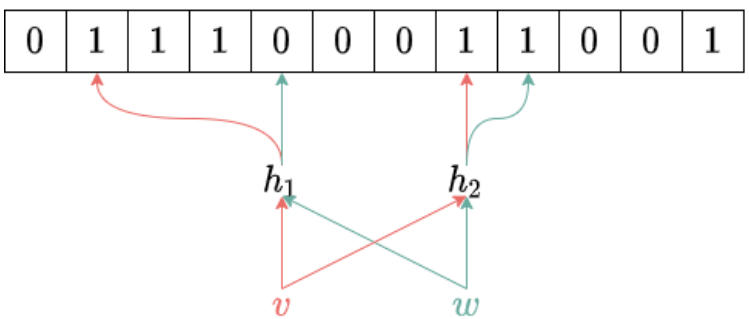


Figura: Esempio dell'operazione di verifica dell'appartenenza.

LBF e SLBF - Introduzione

- Negli anni sono state sviluppate diverse varianti del filtro Bloom, al fine di aumentare ulteriormente l'efficienza della struttura originale. Esempi di queste varianti sono il filtro di Bloom appreso, o learned Bloom filter [3] (LBF), e il sandwiched learned Bloom filter [4] (SLBF).
- Sia LBF che SLBF si caratterizzano per l'utilizzo di un classificatore, al fine di ridurre lo spazio occupato dalla struttura.
- Un $\mathrm{LBF}(g, \tau, B)$ è caratterizzato da un classificatore g, una soglia τ e un filtro di Bloom B, chiamato filtro di backup.
- Un $\mathrm{SLBF}(B_0, g, \tau, B)$ è caratterizzato da un filtro di Bloom iniziale B_0, un classificatore g, una soglia τ e un filtro di backup B.

LBF e SLBF - Controllo dell'appartenenza

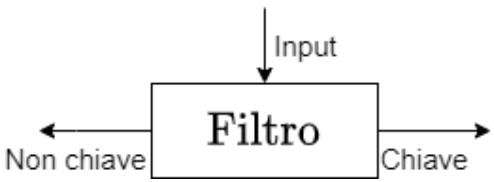


Figura: Verifica dell'appartenenza in un filtro di Bloom.

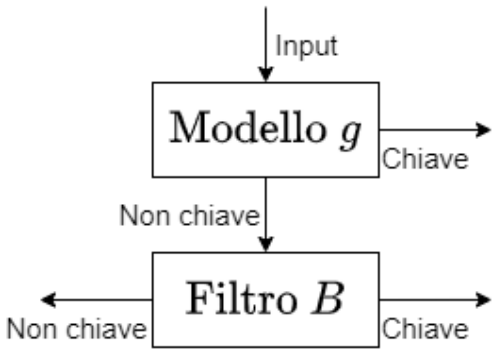


Figura: Verifica dell'appartenenza in un LBF.

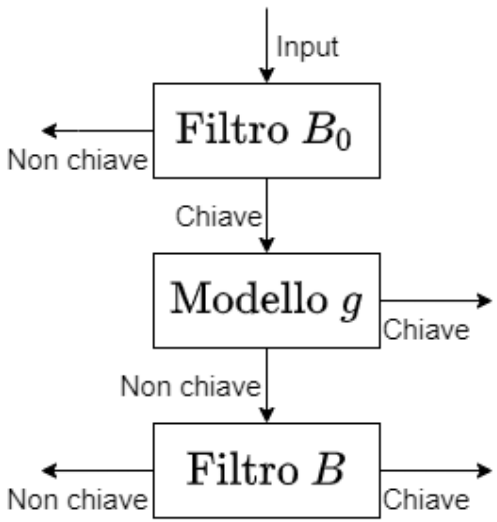


Figura: Verifica dell'appartenenza in un SLBF.

Esperimenti - Introduzione

- Punto di partenza del tirocinio è l'articolo 'An Empirical Analysis of the Learned Bloom Filter and its Extensions' [2], l'obiettivo principale degli esperimenti effettuati è il confronto delle prestazioni dei filtri di Bloom appresi costruiti usando classificatori differenti da quelli dell'articolo.
- Il problema considerato è di classificazione di URL.
- Nello specifico, gli esperimenti effettuati si dividono in due categorie:
  - esperimenti volti a misurare le performance dei classificatori nel problema considerato;
  - esperimenti volti a confrontare le performance dei filtri di Bloom appresi costruiti utilizzando le due tipologie di classificatore.

Esperimenti - Prestazioni dei classificatori

- Esperimenti volti a valutare le prestazioni dei classificatori nel problema di classificazione di URL.
- Si dividono in:
  - Valutazione delle prestazioni della rete ricorrente:
    - grandezza dell'insieme d'addestramento,
    - sbilanciamento del dataset.
  - Valutazione delle prestazioni del percettrone:
    - selezione del modello.
- I risultati hanno evidenziato delle prestazioni migliori da parte del percettrone.

Esperimenti - Analisi dei filtri appresi

- Esperimenti volti ad analizzare empiricamente i filtri di Bloom appresi costruiti usando le due tipologie di classificatore.
- Per entrambe le tipologie di filtro, vengono valutati:
  - Tasso empirico di falsi positivi della struttura.
  - Tasso di falsi negativi del classificatore.
  - Taglia della struttura.
- Vengono valutati diversi valori per la soglia τ variando il rapporto f_τ / f.

Risultati - Analisi empiriche LBF

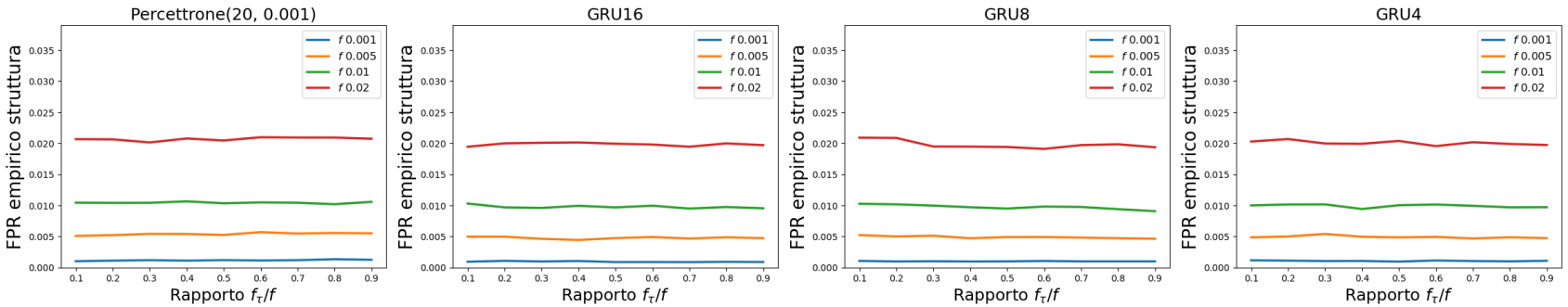


Figura: Tasso empirico di falsi positivi al variare di τ

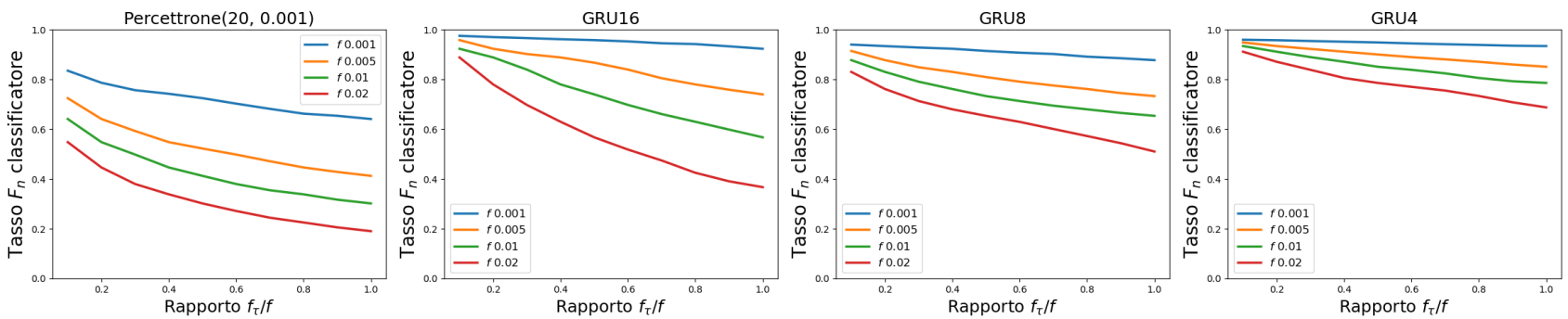


Figura: Tasso di falsi negativi al variare di τ

Risultati - Analisi empiriche SLBF

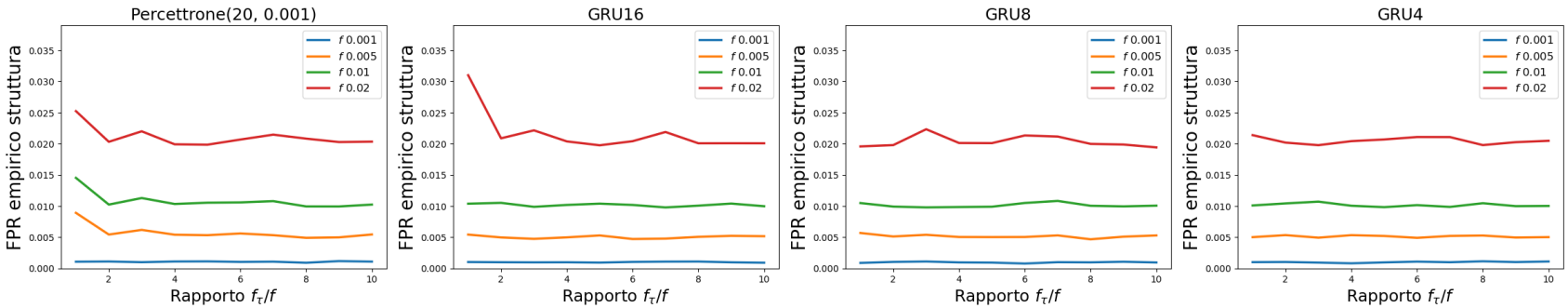


Figura: Tasso empirico di falsi positivi al variare di τ.

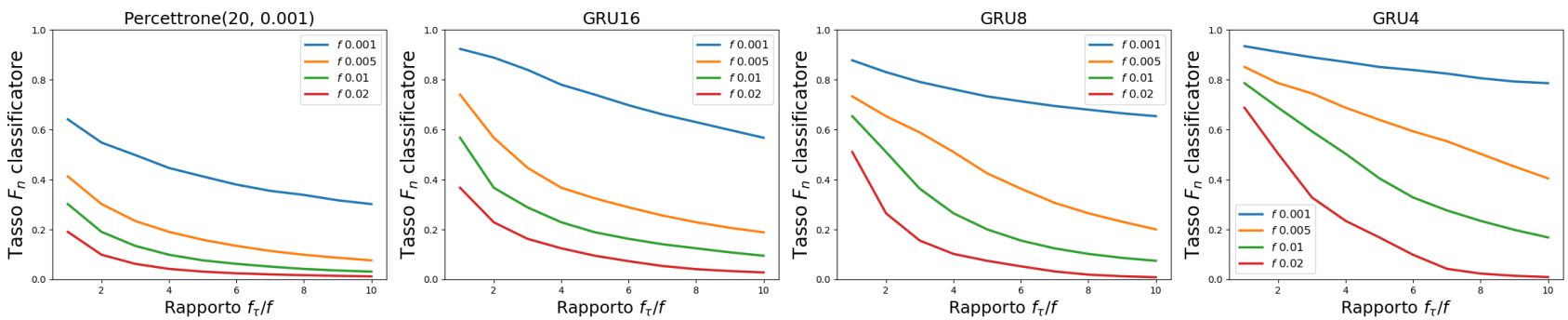


Figura: Tasso di falsi negativi al variare di τ.

Risultati - Spazio occupato

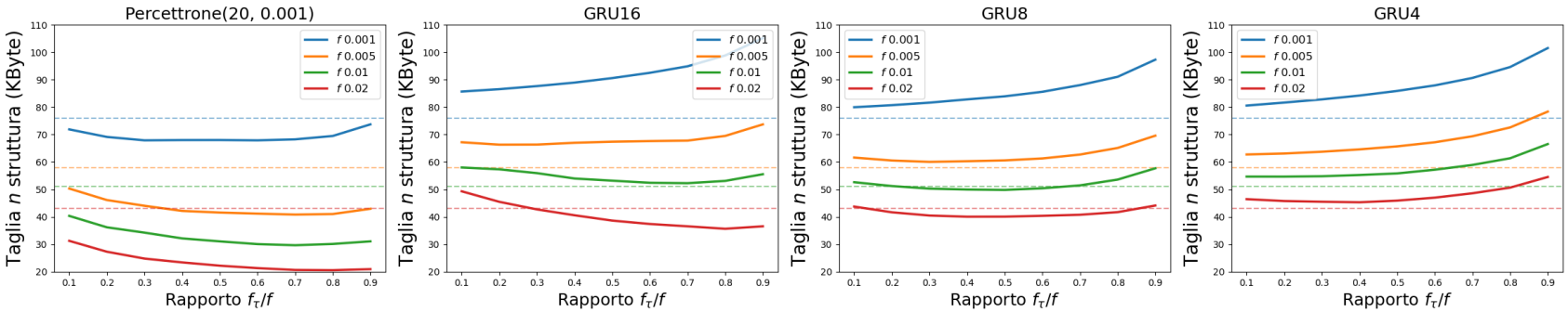


Figura: Spazio occupato dai LBF al variare di τ.

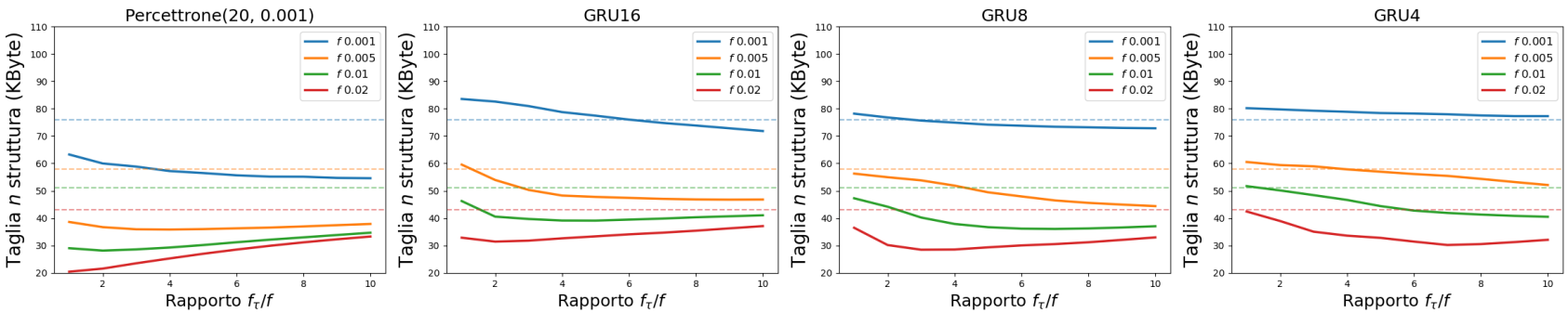
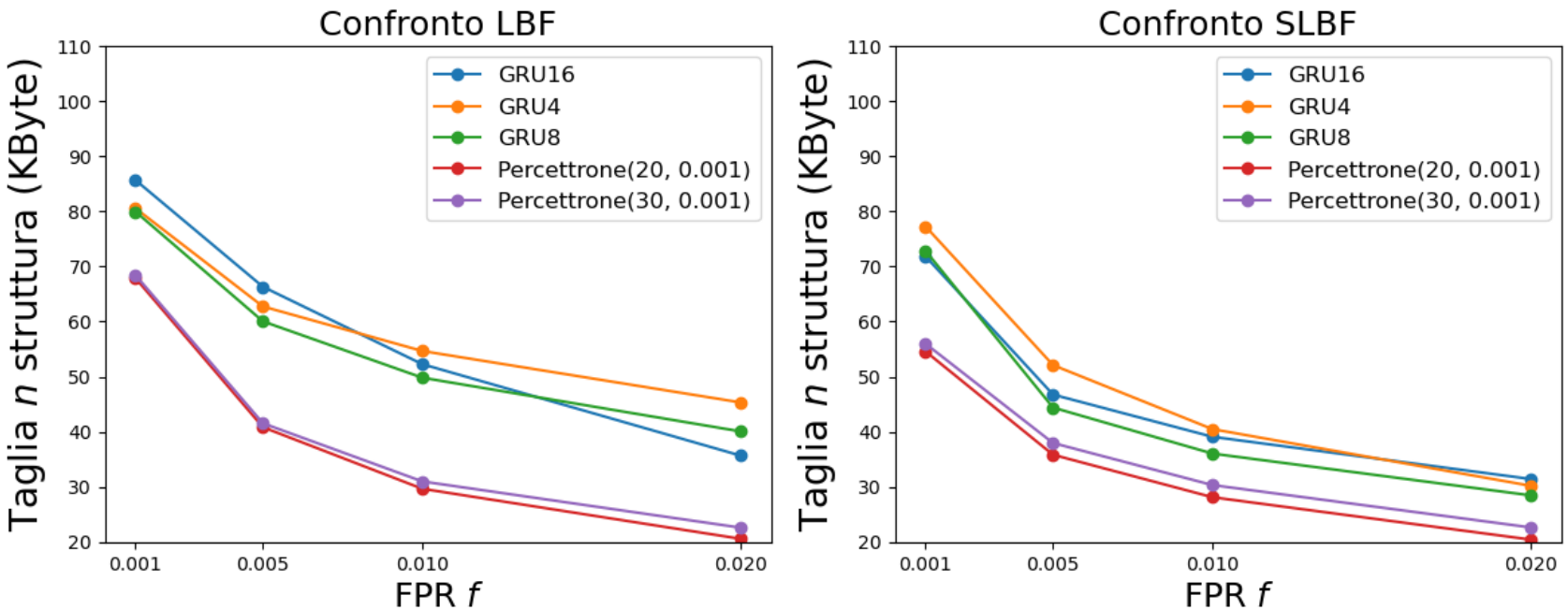


Figura: Spazio occupato dai SLBF al variare di τ.

Risultati - Confronto tra LBF e SLBF



Bibliografia I

[1] B. Bloom. "Space/time trade-offs in hash coding with allowable errors". In: *Commun. ACM* 13 (1970), pp. 422–426.

[2] Celine Liang Jason Ma. "An Empirical Analysis of the Learned Bloom Filter and its Extensions". In: (2020).

[3] Tim Kraska et al. *The Case for Learned Index Structures*. 2018. arXiv: 1712.01208 [cs.DB].

[4] Michael Mitzenmacher. "A Model for Learned Bloom Filters, and Optimizing by Sandwiching". In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. NIPS'18. Montréal, Canada: Curran Associates Inc., 2018, pp. 462–471.