Norme, algoritmi e programmi. Quale differenza?



Periodicamente, quando qualcosa va storto nell'utilizzo dei sistemi informatici in procedure sottoposte a pubblico scrutinio (questo il caso più recente), i media parlano di "discriminazione algoritmica", "dittatura degli algoritmi" e simili espressioni che inventano giornalisti in cerca di un titolo che buca, ma con una, purtroppo, bassa preparazione nella specifica materia. A conferma, consultate per favore la figura 2.5 (p.24) del documento AGCOM "Osservatorio sul giornalismo" del 23 novembre 2020 e leggete il relativo comunicato stampa che recita, testualmente, «*ai professionisti dell'informazione manca un livello di conoscenza specialistica (inteso in particolare in termini di formazione accademica) adeguato alla copertura di fatti ed eventi economici, finanziari, scientifici e tecnologici*». "Ambasciator non porta pena" si dice: spero non me ne vorranno gli amici e le amiche di questo settore, che comunque rispetto.

Chiarito questo punto, vorrei ricordare che nel mondo digitale

il termine algoritmo è di interpretazione lievemente ambigua. Questo perché da un punto di vista storico gli algoritmi sono stati inventati, come metodo, dai matematici (per quei pochi che ancora non lo sanno, ricordo che il termine "algoritmo" deriva dall'appellativo al-Khuwārizmī del matematico arabo del IX secolo Muḥammad ibn Mūsa, che era nativo di Khwarizm, regione dell'Asia Centrale) per descrivere in modo preciso ad altri matematici i procedimenti che dovevano essere utilizzati per la risoluzione dei problemi.

Nell'informatica, questo termine è stato mutuato per descrivere un procedimento risolutivo in modo completamente indipendente da ogni sua realizzazione sotto forma di un programma informatico scritto in un certo linguaggio di programmazione. Il punto di criticità (e la conseguente ambiguità) nasce dal fatto che è il programma informatico (e non l'algoritmo da cui esso viene derivato) quello che viene eseguito dal calcolatore (tecnicamente, dall'automa) e nel passaggio dall'algoritmo al programma, che è una vera e propria traduzione, come accade per i testi letterari, si corre il rischio di "tradire".

Tornando al punto iniziale, è fondamentale tener presente che le norme giuridiche sono già degli algoritmi e, in particolare, sono gli algoritmi che regolamentano ed organizzano la vita sociale: *se trasgredisci questa prescrizione allora ricevi questa punizione*. Le loro macchine esecutrici sono i giuristi: uso liberamente questo termine per parlare dei professionisti della giurisprudenza. In quest'ottica sussiste un parallelo di perfette proporzioni: l'algoritmo sta al matematico come la norma sta al giurista. Non è un caso che sia nel mondo accademico che in quello professionale stanno diventando sempre più numerose figure con una preparazione ibrida tra l'informatica e la giurisprudenza. È un processo analogo a quello che sta già avvenendo da molto più tempo nell'intersezione tra l'informatica e la biologia. Ma questo è un altro discorso.

Nel diritto, tale sovrastruttura sociale di norme scritte e loro esecutori (come la matematica) ha funzionato perfettamente per millenni: il diritto basato su leggi scritte è in uso nel Bel Paese da molti secoli prima che diventasse Nazione. In altre più antiche civiltà ha una storia ancora più lunga. Non ci sono generalmente stati problemi, al di là della naturale tendenza – che possiamo considerare inestricabile dalla natura umana – di alcuni a considerarsi o a valutare il prossimo "più uguale degli altri". Come si dice per la democrazia, anche un sistema sociale basato su un corpus giuridico scritto e su suoi "interpreti" ben preparati, per quanto imperfetto, è migliore di ogni altro sistema.

Quindi, quando norme giuridiche (che sono già algoritmi) vengono trasformate in programmi informatici e nella loro esecuzione qualcosa va storto non è corretto parlare di "errore dell'algoritmo" ma bisognerebbe parlare di "errore di programmazione". Infatti chi esegue il programma informatico, essendo un dispositivo che agisce in modo assolutamente meccanico e completamente predeterminato, non sbaglia mai. Se il risultato ottenuto dall'esecuzione non corrisponde alle aspettative previste nell'algoritmo, l'errore è nel programma informatico. Per continuare nel parallelo con ciò che accade quando gli esecutori sono umani, è come un errore di interpretazione di una norma da parte di un giurista. Solo che, nella stragrande maggioranza dei casi, essendo costui un essere umano dotato di intelligenza, buon senso ed esperienza, possiede una serie di "paracadute" e "cinture di sicurezza" che gli impediscono di stravolgere le norme da interpretare.

È invece molto frequente il caso in cui un insieme di norme, trasformate in un insieme di programmi informatici, diventano qualcosa di lievemente diverso. Vuoi perché questa conversione è fatta in genere da informatici che non posseggono un'approfondita competenza sul tema e ai quali, da parte di chi richiede questa trasformazione, non vengono fornite sufficienti informazioni. Vuoi perché in altri casi le norme,

essendo state scritte pensando ad esecutori umani dotati di intelligenza ed esperienza, non esauriscono tutti i casi possibili. Non ce n'è bisogno, esattamente come accade se chiediamo ad un bambino di andare verso qualcuno: non dobbiamo specificargli di evitare di inciampare in eventuali ostacoli perché è in grado di pensarci da solo. Ecco, gli esecutori dell'informatica, cioè PC, tablet, smarphone e simili, "non ci pensano da soli".

Il problema che sta affliggendo in misura sempre maggiore l'esecuzione automatica tramite programmi informatici di procedure amministrative non è niente altro che l'esempio più recente della difficoltà di realizzare un sistema informatico completamente rispondente alle specifiche, che è il maggior problema ancora aperto dell'ingegneria informatica.

Però, dal momento che tali procedure amministrative sono ormai nella vita quotidiana di tutti noi, queste problematiche – prima ristrette agli esperti di dominio – le stiamo soffrendo ormai tutti.

Parafrasando uno dei finali più famosi della storia del cinema potremmo dire: "Questa è l'informatica, bellezza, l'informatica. E non puoi farci niente. Niente". In realtà, qualcosa possiamo (e dobbiamo) fare. Iniziare ad insegnare l'informatica nella scuola come disciplina scientifica, come si fa per la matematica, per formare generazioni di cittadini in grado di comprendere la differenza tra algoritmi e programmi, e le difficoltà e le insidie che si nascondono nel processo di traduzione.

(I lettori interessati potranno dialogare con l'autore, a partire dal terzo giorno successivo alla pubblicazione, su questo blog interdisciplinare.)

https://www.key4biz.it/norme-algoritmi-e-programmi-quale-differenza/393032/